

---

# A Parallel Trajectory Swapping Wang - Landau Study Of The HP Protein Model

---

A COMPUTATIONAL APPROACH FOR INVESTIGATING LATTICE POLYMERS  
AND THE THERMODYNAMICS OF PROTEIN FOLDING

DEPARTMENT OF PHYSICS SWANSEA, WALES

WRITTEN BY

**Luke Kristopher Davis**

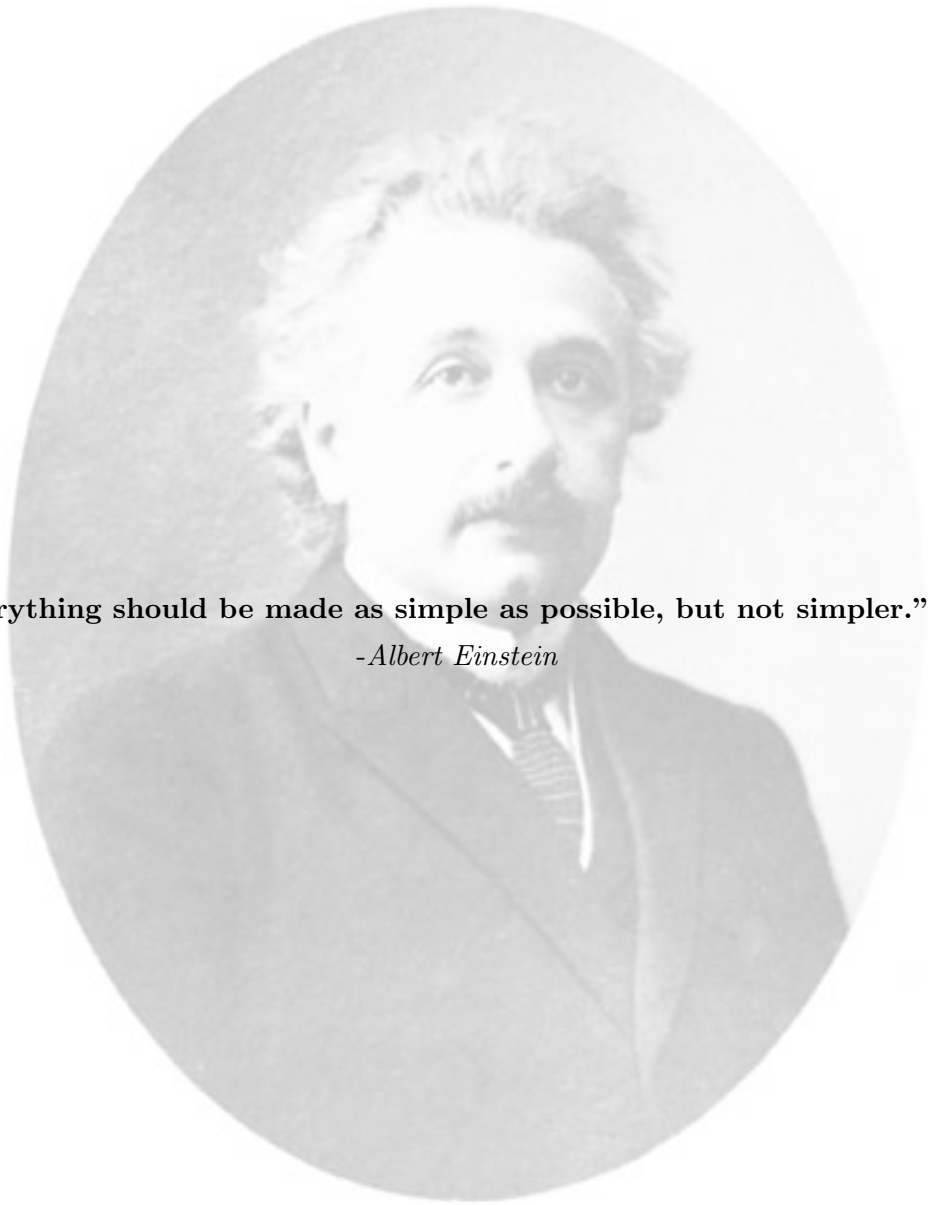
*Supervisor: Professor Biagio Lucini*

2016

FOR THE MPHYS PROGRAMME



**Prifysgol Abertawe  
Swansea University**



**"Everything should be made as simple as possible, but not simpler."**

*-Albert Einstein*

## Abstract

The HP model of protein folding, where the chain exists in a free medium, is investigated using a parallel Monte Carlo scheme based upon Wang-Landau sampling. Expanding on the recent work of Wust and Landau [9] [51] by introducing a lesser known replica -exchange scheme between individual Wang- Landau samplers, the problem of dynamical trapping (spiking in the density of states) was avoided and an enhancement in the efficiency of traversing configuration space was obtained. Highlighting dynamical trapping as an issue for lattice polymer simulations for increasing lengths is explicitly done here for the first time. The  $1/t$  scheme is also integrated within this sophisticated Monte Carlo methodology.

A trial move set was developed which includes pull, bond re-bridging, pivot, kink-flip and a newly invented and implemented *fragment random walk* move which allowed rapid exploration of high and low temperature configurations. A native state search was conducted leading to the attainment of the native states of the benchmark sequences of 2D50 (-21), 2D60 (-36) and 2D64 (-42), whilst attaining minimum energies close to the native state for 2D85(-52 NATIVE= -53), 2D100a (-47 NATIVE= -48) and 2D100b (-49 NATIVE=-50).

Thermodynamic observables such as  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$  were computed for 2D benchmark sequences and folding and unfolding behaviour was investigated. Lattice polymers with monomeric hydrophobic structure were also studied in the same manner with the recording of minimum energy values and thermodynamic behaviour. The native results for the benchmark sequences and lattice polymers were compared with varying computational methods.

---

**Keywords:** HP model, Monte Carlo, Wang-Landau,  $1/t$ , trajectory swapping, protein folding, lattice polymer, thermodynamics, biophysics, dynamical trapping, ISAW, fragment random walk.

# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                                 | <b>10</b> |
| 1.1      | Protein Structure and Function . . . . .            | 10        |
| 1.2      | One or Many Driving Forces? . . . . .               | 12        |
| 1.3      | The Protein Folding Problem . . . . .               | 14        |
| 1.4      | Introduction to the HP Model . . . . .              | 15        |
| 1.4.1    | Introduction to The Self-Avoiding Walk . . . . .    | 17        |
| 1.5      | What has been done already? Computational . . . . . | 18        |
| 1.5.1    | The Work of Wust and Landau . . . . .               | 20        |
| 1.6      | Aims of this project . . . . .                      | 21        |
| <b>2</b> | <b>Necessary Theory</b>                             | <b>22</b> |
| 2.1      | Statistical Mechanics . . . . .                     | 22        |
| 2.1.1    | Canonical Ensemble . . . . .                        | 22        |
| 2.1.2    | Energy Fluctuations and Observables . . . . .       | 23        |
| 2.2      | Probability Theory . . . . .                        | 24        |
| 2.2.1    | Markov Chains . . . . .                             | 24        |
| 2.2.2    | Non- Markovian Schemes . . . . .                    | 24        |
| 2.2.3    | Ergodic Process . . . . .                           | 24        |
| 2.2.4    | Ergodic Hypothesis . . . . .                        | 25        |
| <b>3</b> | <b>Methodology</b>                                  | <b>26</b> |
| 3.1      | Monte Carlo Methods . . . . .                       | 26        |
| 3.1.1    | Wang Landau Sampling . . . . .                      | 26        |
| 3.1.2    | 1/t algorithm . . . . .                             | 28        |
| 3.1.3    | Detailed Balance . . . . .                          | 30        |
| 3.2      | Lattice System . . . . .                            | 31        |
| 3.3      | Dynamical Trapping . . . . .                        | 32        |
| 3.4      | Logistics of the parallel implementation . . . . .  | 34        |
| 3.5      | Trial Move Implementation . . . . .                 | 35        |
| 3.5.1    | Fragment Random Walk . . . . .                      | 40        |
| 3.5.2    | LCSAW and Excluded Volume Barriers . . . . .        | 42        |
| 3.5.3    | Trial Move Testing . . . . .                        | 42        |
| 3.6      | Energy Computing Routine . . . . .                  | 45        |
| <b>4</b> | <b>Energy Interval Experiment for WLS</b>           | <b>46</b> |
| 4.1      | Discussions and Remarks . . . . .                   | 46        |
| <b>5</b> | <b>Results</b>                                      | <b>49</b> |
| 5.1      | Native State Search . . . . .                       | 49        |
| 5.2      | Wang Landau Sampling . . . . .                      | 51        |

|           |   |           |
|-----------|---|-----------|
| 5.2.1     | 2D50 . . . . .  | 51        |
| 5.2.2     | 2D60 . . . . .  | 53        |
| 5.2.3     | 2D64 . . . . .  | 55        |
| 5.2.4     | 2D85 . . . . .  | 57        |
| 5.2.5     | 2D100a . . . . .  | 59        |
| 5.2.6     | 2D100b . . . . .  | 61        |
| 5.3       | ISAWs . . . . .   | 63        |
| 5.3.1     | Error Analysis . . . . .                                | 68        |
| <b>6</b>  | <b>Discussion of Results</b>                            | <b>69</b> |
| 6.0.2     | Native State Search . . . . .                           | 69        |
| 6.0.3     | Thermodynamic Investigations . . . . .                  | 71        |
| <b>7</b>  | <b>Conclusions</b>                                      | <b>76</b> |
| <b>8</b>  | <b>Areas for Future Work</b>                            | <b>77</b> |
| <b>9</b>  | <b>Acknowledgements</b>                                 | <b>78</b> |
|           | <b>Appendices</b>                                       | <b>79</b> |
| <b>A</b>  | <b>Amino acid HP table</b>                              | <b>79</b> |
| <b>B</b>  | <b>Preliminary Testing Results</b>                      | <b>79</b> |
| B.1       | Trial Move Prioritising . . . . .                       | 79        |
| B.2       | Energy Scoring . . . . .                                | 80        |
| B.3       | Results . . . . .                                       | 82        |
| <b>C</b>  | <b>Replica Exchange Routine</b>                         | <b>84</b> |
| <b>D</b>  | <b>Critical temperatures for 2D benchmark sequences</b> | <b>85</b> |
| <b>10</b> | <b>References</b>                                       | <b>86</b> |

## List of Figures

|    |   |    |
|----|---|----|
| 1  | What defines the amino acid is its side chain (blue). ( <i>Thanks go to Nature Education in Protein Structure</i> ) . . . . .   | 11 |
| 2  | The hydrophobic material is surrounded by a three dimensional cage of bonded water molecules called clathrate structures. . . . .   | 11 |
| 3  | A diagram showing the basic structure and atomic composition of an alpha helix with H-bonds highlighted. [2] . . . . .  | 13 |
| 4  | Tertiary structure determined by the lowest energy state computed in a multicanonical Monte Carlo study (black) superposed with structure found from X-ray crystallography (grey). [21] . . . . .   | 13 |
| 5  | The protein traverses the energy landscape attaining a more stabilized structure as it ventures towards the global native minimum. Note the 'string-like' configurations at higher energies and the compact native structure at the minimum. . . . .  | 14 |
| 6  | A 2D native state of the protein sequence S1-8 (2D64) in the 2D HP model found by the ACO method. Black beads are hydrophobic amino acids and white beads are polar. Thick black lines represent the covalent bonds between sequence amino acids. [46] . . . . .  | 16 |
| 7  | The computation of observables at different temperatures for sequence 2D100a. Specific heat capacity $C/N$ is shown in black (left ordinates), root mean squared radius of gyration $R_g/N$ is shown in red (outer right ordinates) and tortuosity $\tau$ is shown in blue (inner right ordinates). . . . .                     | 20 |
| 8  | Spiked DOS for (a) the frustrated XY model and (b) the 8-mer poly-alanine. For details see [44]. . . . .  | 33 |
| 9  | The numbers represent the thread ID's and letters represent arbitrary configurations. Trajectories are swapped through random shuffling. The arrow to the right represents the direction of MC time. . . . .  | 34 |
| 10 | An example of an end-bond flip move where the penultimate monomer or second monomer acts as an axis (blue) so that the 1st or last monomer can rotate about it. . . . .   | 35 |
| 11 | An example of a kink flip move where the white monomer is at a corner between its sequential neighbours and 'flips' to the opposite corner if it is free. . . . .   | 35 |
| 12 | An example of a pull move where the anchor monomer (blue) remains fixed and the primary monomer (purple) will move to $P$ and the secondary monomer (green) moves to $G$ . The rest of the chain 'slithers' behind to keep the sequence and length of the chain fixed. . . . .  | 36 |
| 13 | An example of a $\odot$ pivot move where the anchor monomer (blue) remains fixed and the red and grey monomers move according to the change in directionality relative to the preceding monomer. Initially red was to the right of blue and now it is below it also grey was below red and now it is to the left of it. . . . . | 36 |

|    |   |    |
|----|---|----|
| 14 | The four possible scenarios for a kink flip move, the orange $O$ represents the future position of the primary monomer (orange). From left to right the names of the moves are as follows: <i>bottom left quadrant move</i> , <i>top right quadrant move</i> , <i>top left quadrant move</i> and <i>bottom right quadrant move</i> . . . . .            | 37 |
| 15 | The red crosses represent bonds that will be destroyed and dotted lines represent future connected bonds. $k$ and $p$ are monomer values within the set $\{1, \dots, N\}$ (Ignoring other monomers for clarity). . . . .  | 39 |
| 16 | An example of a type II bond-rebridging move on a small lattice protein chain. Note the re-uploading of the HP sequence. . . . .  | 40 |
| 17 | 1) The initial configuration with a randomly selected fragment. 2) Fragments are being produced at random, any fragments which do not connect to the fixed points are rejected. 3) A successfully regrown fragment and its resulting configuration. ( <i>Picture originally published in [36] and gratitude goes to Zhang, Kou and Liu.</i> ) . . . . . | 41 |
| 18 | 1) The initial configuration before the FRW move. 2) A fragment is chosen. 3) The resulting configuration of a successful FRW move. . . . .   | 41 |
| 19 | The evolution of this chain goes from left to right. (H) monomers are represented as black circles and (P) monomers are represented as white circles. The green dashed lines represent H-H contacts and for this chain represent native states, since the maximum number of H-H contacts is 1. . . . .  | 43 |
| 20 | The 10-mer before any moves (left) and after 500 pivotmoves (right). The HP sequence (HHPPHHPPHH) of monomers remain invariant and the chain remains intact which means the moves respect the conditions of the HP model and LCSAW. . . . .   | 43 |
| 21 | The first five configurations representing the evolution of the 6-mer via pull moves. One can see that three unique native states have been found. . . . .  | 43 |
| 22 | The 10-mer before any moves (left) and after 500 pull moves (right). The HP sequence (HHPPHHPPHH) of monomers remain invariant and the chain remains intact which means the moves respect the conditions of the HP model and LCSAW. . . . .   | 44 |
| 23 | Simple chain pathway from linear chain $\rightarrow$ pull moved chain $\rightarrow$ kink of chain being flipped successfully. . . . .   | 44 |
| 24 | A longer sequence of successful pull and kink flip moves. . . . .   | 44 |
| 25 | The starting linear chain (left) and the resulting chain (right) after 1000 moves. . . . .  | 45 |
| 26 | Purple = run 1, green = run 2, light blue = run 4 and gold = run 5. The error in $C_v$ is the final modification factor shown in table 5, the errors for run 2 and 4 were omitted since they are smaller than the data points. . . . .  | 47 |
| 27 | Purple = run 1, green = run 2, light blue = run 4 and gold = run 5. The error in $S$ is the final modification factor shown in table 5, the errors for run 2 and 4 were omitted since they are smaller than the data points. . . . .  | 48 |

|    |  |    |
|----|--|----|
| 28 | Example native structures. . . . .   | 50 |
| 29 | Computed thermodynamic observables for 2D50. . . . .   | 52 |
| 30 | Computed thermodynamic observables for 2D60. . . . .   | 54 |
| 31 | Computed thermodynamic observables for 2D64. . . . .   | 56 |
| 32 | Thermodynamic observables for 2D85. . . . .  | 58 |
| 33 | Thermodynamic observables for 2D100a. . . . .  | 60 |
| 34 | Thermodynamic observables for 2D100b. . . . .  | 62 |
| 35 | Specific heat capacity per monomer, $C_v/N$ , against temperature $T$ . Length 25 (purple, green error bars) = lower curve, length 64 (green, blue errorbars) = middle curve and length 100 (black, purple error bars) = highest curve. . . . .  | 65 |
| 36 | Internal energy per monomer, $U/N$ , against temperature $T$ . Length 25 (blue error bars), length 64 (green errorbars) and length 100 ( purple error bars) = highest curve. . . . .   | 66 |
| 37 | Graphical comparison of minimum energies found for benchmark ISAWs. N.B. simulation timings for the Wust-Landau results unknown. 'isawnative' are results from this work and 'wangisaw' are results taken from [51]. . . . .   | 67 |
| 38 | Comparison of native structures for 2D64. The similarity of the external polar amino acid placement is striking, also each sequence has the same line of symmetry (externally). The difference between the native structures is found within the hydrophobic core, however this will not necessarily alter the function of the protein since it interacts with others via its external structure. . . . .  | 70 |
| 39 | Diagram reflecting a rough (simplistic) 2D energy folding funnel of an arbitrary protein. At higher temperatures and energies the protein becomes denatured (unfolded) and entropy dominates. As $T \rightarrow 0$ the protein forms a molten globule and then enters the near native region. The native state is located exactly at the minimum of the energy folding funnel. The transition temperature $T_C$ can be located anywhere between the molten globule and native state. . . . . | 73 |
| 40 | Notice the similarity in values and qualitative behaviour. One noticeable difference is the 'rough' quality the $C_V/N$ from Wust and Landau has in the native region whereas my results are smoother. This could reflect that Wust and Landau ran their simulations for longer and hence sampled the conformational space in the native region more thoroughly. . . . .   | 74 |
| 41 | The ratio genus/energy of a homopolymer on a cubic lattice, as a function of $T$ , at different lengths of the chain. Thanks go to the authors of [52]. . . . .  | 75 |
| 42 | The routine in main which attempts 5000 moves on the chain recording minimum energy configurations. . . . .  | 81 |
| 43 | The configurations (i),(ii) and (iii) found in the 2D7A runs. . . . .  | 82 |



## List of Tables

|    |  |    |
|----|--|----|
| 1  | Contributions to the energy. . . . .   | 16 |
| 2  | An example 2D lattice with locations stored in a 1D array. . . . .   | 31 |
| 3  | An example of (H) residues on a 2D lattice of side length $L = 4$ . . . . .  | 32 |
| 4  | How relative direction is changed under the two rotations. . . . .   | 37 |
| 5  | For the energy ranges 0 is the upper bound also note that $\ln[f_{initial}] = 1$ as outlined in section 3.1. . . . .   | 46 |
| 6  | Comparison of native states found in this work (blue) with different methods .   | 49 |
| 7  | The right column reflects the convergence of the intrinsic DOS for each process, the majority are $\leq 10^{-7}$ , this convergence is adequate for the results shown in figure 29 . . . . .             | 51 |
| 8  | The right column reflects the convergence of the intrinsic DOS for each process, the majority are $\leq 10^{-30}$ , this convergence is adequate for the results shown in figure 30. . . . .             | 53 |
| 9  | The right column reflects the convergence of the intrinsic DOS for each process. This convergence is questionably adequate for the results shown in figure 31 (see section 6 for an explanation. . . . . | 55 |
| 10 | The right column reflects the convergence of the intrinsic DOS for each process, the majority are $< 10^{-5}$ , this convergence is adequate for the results shown in figures 32. . . . .                | 57 |
| 11 | The right column reflects the convergence of the intrinsic DOS for each process, the majority are $< 10^{-5}$ , this convergence is adequate for the results shown in figure 33 . . . . .                | 59 |
| 12 | The right column reflects the convergence of the intrinsic DOS for each process, the majority are $< 0.01$ . Observables for this run are shown in figure 34. . . .                                      | 61 |
| 13 | Final modification factors for ISAW length simulations. . . . .  | 64 |
| 14 | Monte Carlo iterations and duration of simulation runs. . . . .  | 64 |
| 15 | Configuration type (i), (ii) and (iii) are shown in figure 43. . . . .   | 82 |
| 16 | Results for 2d10a . . . . .  | 83 |
| 17 | Results for 2d20a . . . . .  | 83 |

# 1 Introduction

Evolution has, through billions of years of selection mechanisms, formed a vast array of living organisms on this planet [1]. These organisms have intricate internal machinery which allows them to persist through time and compete to pass their genetic information on to the next generation. Proteins are the main workers in all living organisms which fuel this internal machinery.

Proteins are the building blocks of cells and they also perform nearly all the cell's functions. For instance, enzymes provide the molecular surfaces in a cell that promote its multitude of chemical reactions [2]. Some proteins send messages from one cell to another which is vital for large scale cellular activity. Yet others act as tiny molecular machines with moving parts [2] *kinesin*, for example, allows organelles to travel through the cytoplasm via propulsion; also *topoisomerase* can unravel knotted DNA molecules.

The physics of proteins, ranging from folding mechanisms to calculations of specific binding energies of ligands, has developed rapidly over the last 50 years [17] and is of great interest to computational physicists and those from a statistical mechanical background. The development of the HP model of proteins devised by K.Dill [5], outlined in 1.4, which provides a simple 'Ising-like' model has enabled scientists to use computational techniques to explore the global transitions of proteins into their native state.

It is imperative that scientists build a solid and comprehensive understanding of proteins in order to paint a complete picture of the mechanisms of life.

## 1.1 Protein Structure and Function

Proteins are chains which contain sequences of amino acids which, when one considers the protein as a polymer, act as the repeating subunits known as monomers. These monomers connect to one another via a peptide bond. There are 20 known amino acid bases which form an alphabet (see Appendix A) and the sequence of a protein chain consists of elements within this alphabet. An amino acid is a chemical group which is defined by its chain residue, differing residues have different chemical properties.

These small organic molecules (amino acids) consist of an alpha carbon atom connected to an amino group, a carboxyl group, a hydrogen atom and a variable side chain as shown in figure 1.

Residues can be hydrophobic or polar (hydrophilic) (or degrees of both) <sup>1</sup>, vary in size and charge [25].

The structure of a protein can be described in the following hierarchical way; **primary structure**: the sequence of amino acid bases, **secondary structure**: local formation of  $\alpha$  helices and  $\beta$  sheets, **tertiary structure**: typically the 3 dimensional structure of a protein domain in the native structure which is more irregular than the secondary structures [4] and

---

<sup>1</sup>Hydrophobic effect arises from the fact that water molecules seek to form hydrogen bonds with each other and push non-polar material away to form these bonds. Polar material can form hydrogen bonds [4]

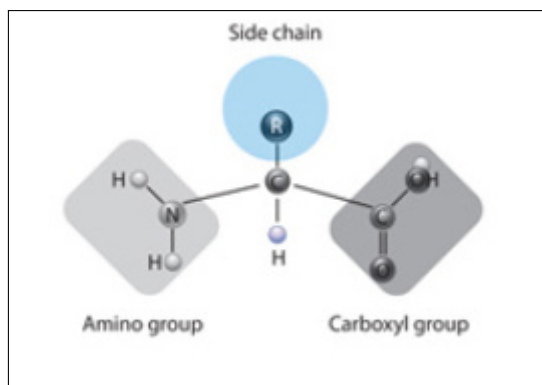


Figure 1: What defines the amino acid is its side chain (blue). (*Thanks go to Nature Education in Protein Structure*)

the **quaternary structure**: the 3 dimensional, native structure of the fully functional protein [25].

It is known that the sequence of amino acids determine the three dimensional structure of the protein which affects how it interacts with other molecules [25] [4][2]. A protein molecules physical interaction with other molecules determines its biological function [2]. For example, antibodies in the human immune system recognize antigens by having a complementary surface to that of the antigen [25]. Also the enzyme *hexokinase* binds glucose and ATP so as to catalyze a reaction between them.

All proteins bind to other molecules, where in some cases the binding is strong and in others weak. The binding always shows great specificity.

## Hydrophobic Effect

It has been mentioned that a subset of amino acid bases are hydrophobic, which means they cannot form bonds with the surrounding water molecules, hence the water molecules prefer to bond with themselves and the hydrophilic bases. The water pushes these hydrophobic bases away in order to form these preferred bonds. This pushing is a direct consequence of the water molecules forming an ice-like structure around the hydrophobic material which drives the protein into a compact structure ( *See figure 2* ).

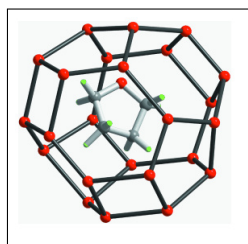


Figure 2: The hydrophobic material is surrounded by a three dimensional cage of bonded water molecules called clathrate structures.

## Local Forces

There are forces which occur between the atoms and molecules of proteins and hence should, in principal, physically play a role in the folding mechanism and stability of the structure. *Covalent bonds* involve the sharing of two electrons between the interacting partners. For example, the  $H_2O$  water molecule has its hydrogen atoms bound to the oxygen atom via covalent bonding. The bonding leaves  $H^+$  with an excess positive charge, and  $O^-$  with an excess negative charge.

*Hydrogen bonds* involve sharing an H atom between the interacting partners. The bond has polarity with H covalently bonded to one partner and more weakly attached to the other through its excess charge [4]. *Ionic bonds* arise from the exchange of one electron. There are also *Van Der Waals* interactions which arise from temporary mutual electric polarization. Since molecules can have charge they must interact through the coulomb potential which is screened by the surrounding aqueous solution.

### 1.2 One or Many Driving Forces?

It was accepted that the mechanism of protein folding was a sum of the contributions of different local interactions as briefly described in section 1.1. The prevailing paradigm of the folding sequence asserted that the primary structure encoded the secondary structure which then determined the tertiary structure [14].

Sophisticated statistical mechanical simulations have unearthed a new view on the dominant driving component of protein folding. Making varying use of the HP model for proteins these simulations show that the hydrophobic effect outlined in section 1.1 is the dominant driving force and its effects, while non-specific in nature, are felt locally and non-locally in the sequence [9] [15].

Electrostatic interactions among the charged side chains are not likely to dominate the folding process. This is because most proteins have few charged residues which are concentrated in high-dielectric regions on the protein surface [17]. Hydrogen bonding is a key element in the formation of the secondary structures in a protein state, for example hydrogen bonds between oxygen and hydrogen help to form the  $\alpha$  helix (*see* figure 3). Also when the protein becomes increasingly more compact, Van der Waals interactions described in section 1.1 play a significant role [16].

However there is greater interest and importance attached to finding the dominant factor which distinguishes how two separate proteins fold into distinct native structures. There is considerable evidence, experimental and computational, that shows that the hydrophobic effect is the dominant driving force for the folding of proteins.

For example model compound studies show 1-2 kcal/mol for transferring a hydrophobic side chain from water into oil-like media and there are a significant amount of them [20] [17].

Sequences that keep there HP sequence but have there amino acids jumbled fold to their respective native conformations without the need to tamper with local interactions [17] [and

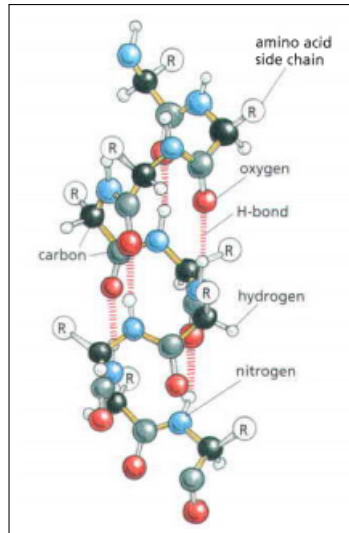


Figure 3: A diagram showing the basic structure and atomic composition of an alpha helix with H-bonds highlighted. [2]

references therein].

Also computational simulations using the HP model have reproduced tertiary structures of proteins very well, for example the tertiary structure of the C-peptide of ribonuclease A (*see* figure 4) [6].

For free energy and thermodynamic calculations on simple HP models on square 3D lattices have also proven very successful [9] [6]. Hence simulations and empirical investigations focusing on the hydrophobic nature of the bases to probe the global behaviour of folding are well founded.

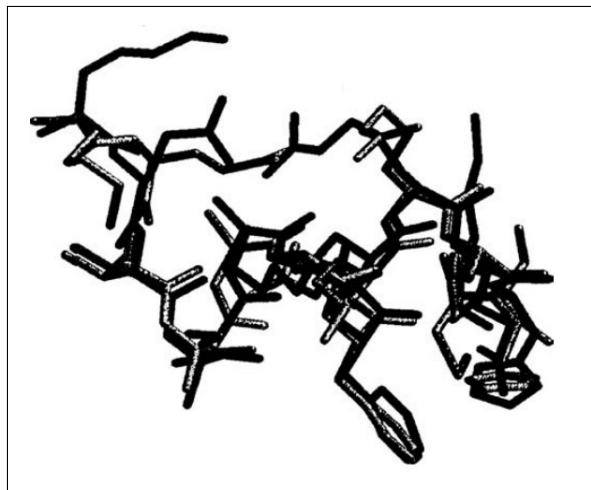


Figure 4: Tertiary structure determined by the lowest energy state computed in a multicanonical Monte Carlo study (black) superposed with structure found from X-ray crystallography (grey). [21]

## Stabilization of secondary structures

Studies of proteins, namely of the lattice and tube variety, have revealed that secondary structures of the protein are stabilized due to the compactness of the conformation which is a direct consequence of the hydrophobic effect in action [17].

## Folding Into The Native State

Proteins fold into conformations which minimize the entropy, they are guided into this structure by the non-local hydrophobic force and the secondary, tertiary and quaternary structures were thought to be stabilized solely by local hydrogen bonding [2]. However, it has been argued [17] that the secondary structures become more stabilized as the protein forms a tighter conformation and hence the tertiary structure directly controls this.

The energy landscape of proteins is normally rough and complex, hence there is no absolute minimum but rather a group of minima or constraint minimum which define the preferable conformations of proteins [4]. As the protein changes conformation its energy states glide along the energy landscape and are guided towards the most stable state (*see figure 5*). The protein is encouraged into this native state due to the aqueous solution surrounding it which sparks the hydrophobic force to act. Each protein normally folds up into a single stable conformation.

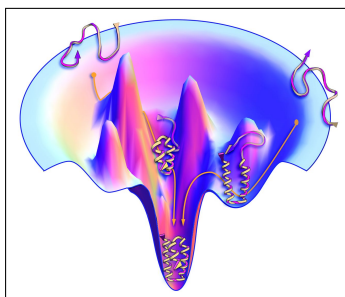


Figure 5: The protein traverses the energy landscape attaining a more stabilized structure as it ventures towards the global native minimum. Note the 'string-like' configurations at higher energies and the compact native structure at the minimum.

Due to interactions with other molecules in the cell the native conformation changes. These changes in structure however are usually crucial to how the protein functions.

### 1.3 The Protein Folding Problem

The birth of a protein folding problem arose, in 1961, from the experimental results of Anfinsen on ribonuclease [13]. The conclusions drawn from these results show that the sequence of amino acids is enough information to dictate the native conformation of the protein in a specific solution. The native structure is then indifferent to how the polypeptide chain is synthesized in the first place, say if it was synthesized on a ribosome or in a test tube [17].

This spurred biologists and biochemists to conduct experimental work on the amino acid

sequence in light of the fact that a protein in a test tube environment could convincingly replicate its behaviour in an organism (there are rare exceptions for example see [17]).

Following these results one can then ask: what thermodynamic and kinetic interactions take the string like protein with its amino acid sequence to a compact native structure? Various approaches, experimental and computational, have been devised to tackle this question. For example NMR (nuclear magnetic resonance) imaging is used to probe the details of folding and misfolding<sup>2</sup>, allowing characterization of the molecular structure and dynamics of folding.

Molecular dynamics simulations, which invoke various force fields to mimic inter-atomic forces (see 1.1) and Monte Carlo methods are recently proving successful in characterising properties of folding [9] [18].

However there is no unified framework, analytical or computational, which can adequately describe and explain the complete mechanisms of protein folding.

It is of my opinion that there are essentially two problems that the scientific community needs to address in order to form a complete understanding of protein folding.

**Problem A :** What are the *detailed* physical mechanisms, atomistic and statistical mechanical, of protein folding?

**Problem B :** Can we efficiently and consistently predict, via computational simulation, the native structure of any amino acid sequence?

Problem A appeals to our desire to understand the basic mechanisms of nature whilst problem B is more focused on medical, biological and pharmaceutical applications. A solution to problem B in the form of a universal computer program (UCP) could allow the quick prediction of new proteins from amino acid sequences which are not found in nature. This could lead to artificial proteins bettering those carved out by natural selection and further advance the battle with dominant diseases.

It is obvious that these two problems are not independent and that success in problem A will further success in problem B and vice versa. As resources and techniques in high performance computing have, without doubt, increased in power we will no doubt see rapid progress on these problems.

## 1.4 Introduction to the HP Model

Inspired by the accumulation of evidence affirming that the hydrophobic effect is the globally dominant driving force in the folding process for globular proteins K.A Dill proposed a simplified model to characterize this striking behaviour (*a review is given by [5]*). He proposed that the alphabet of 20 amino acids should each be labelled as 'hydrophobic' or 'polar' (see appendix

---

<sup>2</sup> Misfolding of proteins occurs when the protein does not find its most thermodynamically stable state but is fixed in its partially folded state by thermodynamic means or interactions with other molecules. The misfolding of proteins leads to modified functionality which can be toxic [19].

A for a conversion table for all the amino acids). Then the monomeric sequence of the protein is a sequence of (H)'s and (P)'s.

There are four simple rules for a HP protein:

1. Monomers have uniform size.
2. The peptide bond length between monomers is uniform.
3. Positions of the monomers are restricted to positions on a regular lattice.
4. No two monomers can occupy the same position and overlapping of bonds is forbidden.

There are only nearest-neighbour interactions and there is an attractive potential  $\epsilon_{HH}$  between two H monomers which are topologically connected i.e. not direct neighbours on the sequence.

Hence the Hamiltonian of this simple system is given by:

$$H = -\epsilon_{HH}n_{HH} \quad (1)$$

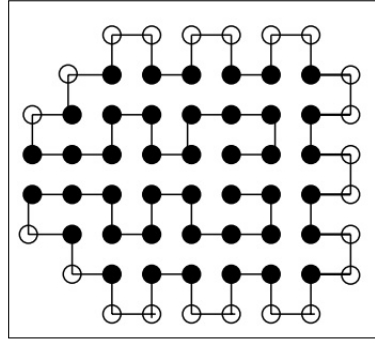


Figure 6: A 2D native state of the protein sequence S1-8 (2D64) in the 2D HP model found by the ACO method. Black beads are hydrophobic amino acids and white beads are polar. Thick black lines represent the covalent bonds between sequence amino acids. [46]

Where  $n_{HH}$  is the number of nearest neighbour topologically connected H-H monomers. The general energy function has values obeying table 1.

|   | H                | P |
|---|------------------|---|
| H | $-\epsilon_{HH}$ | 0 |
| P | 0                | 0 |

Table 1: Contributions to the energy.

A native state is a conformation having a minimum Hamiltonian energy value. Despite the simplicity of the model it is difficult, for long chain lengths, to compute the lowest possible energy for the folded chain [12].



In 2D and more especially in 3D one of the drawbacks of this model is that it is highly degenerate. This is emphasized for long sequences of protein chains. The degeneracy is normally very low in the low temperature ranges [9].

#### 1.4.1 Introduction to The Self-Avoiding Walk

The conditions of the protein chain in the HP model outlined in section 1.4 exactly match the conditions of a length conserving self-avoiding walk (LCSAW)[4][9][12][25].

In general, A self-avoiding walk is a path on a lattice that does not visit the same site more than once [3]. It sits on an undirected graph which is a collection of points, with a collection of pairs of points known as *edges*. The basic undirected graph which is used here and in the literature is the *d-dimensional hypercubic lattice*  $Z^d$ . The points of this lattice are of the *d*-dimensional Euclidean space  $R^d$  in which all the components are all integers, and the edges are given by the set of all unit line segments as *nearest-neighbour bonds*. The LCSAW is defined formally as follows:

**Definition (LCSAW)** Let  $d \geq 1$ . An  $n$  - step *self-avoiding walk* from  $x \in Z^d$  to  $y \in Z^d$  is a map  $w: [0, n] \rightarrow Z^d$  with:

1.  $w(0) = x$  and  $w(n) = y$
2.  $|w(i+1) - w(i)| = 1$  (unit length bonds)
3.  $\forall i, j \in [0, n], i \neq j \Rightarrow w(i) \neq w(j)$
4.  $|w|$  is a constant.

The main idea here is that the movements of the protein chain respect the self-avoiding conditions and each move which obeys these generates a new LCSAW.

While physicists and biologists using the HP model merely make use of the properties of SAWs and algorithms for move sets on them, the mathematics behind SAWs is rich and contains many open basic questions. For a rigorous introduction and overview see Madras and Slade [3].

#### NP Completeness

**Problem B**, which is the problem of predicting the native conformation of a protein chain defined by a sequence of amino acids, can be stated formally as a combinatorial optimization problem in the HP model [24]:

**Optimal Folding Problem:** Given a sequence of (H)'s and (P)'s, find a LCSAW on the 2D or 3D lattice which maximises  $n_{HH}$  (the number of H-H contacts).

It has been proven that this problem is NP - complete in 2D and 3D [22]. Their proof revolves around asking whether the graph representing the HP model contains a Hamiltonian

cycle. This means that finding the conformation which minimizes the energy cannot be done in polynomial time.

The proof, although initially far removed from proteins, does emphasize the need for the protein chain to form a compact cubic shape in 3D [22].

There is an interesting question whether all HP models on all lattices are NP-complete, since it has not been formally shown that the problem is NP-complete for triangular or non-square lattices and that different computational methods may affect the exact nature of the problem [23].

### **The need for computational studies**

Recalling the two specific sub-problems associated with the general protein folding problem stated in 1.3 it appears that both of them cannot be solved analytically. Since finding native conformations has been mathematically proven to be NP-hard and that the complicated nature of atomistic dynamics seems to evade purely pen and paper advances, it seems inevitable that the community will need to make use of sophisticated computational simulations.

This not only means carefully constructed models, simulation techniques and software but advances in computational hardware are needed to meet the computational demands.

Also as protein folding and protein structure prediction are interdisciplinary areas of study, it is necessary for those using computational techniques to directly work in unison with those conducting experiments.

## **1.5 What has been done already? Computational**

### **Molecular Dynamics**

Molecular dynamics (MD) concerns itself with simulating the physical system by keeping track of all the coordinates of the constituent particles. The system then evolves in time obeying equations of motion (usually Newtonian) which are integrated numerically. Simulating a protein molecule, which is a macroscopic system relative to simple atomic systems, is extremely computationally demanding if one uses atoms as the basic constituent particles. Hence coarse grained approaches are used. For example the Go model is a popular coarse graining approach where the protein is represented as a chain of one-bead amino acids whose structure is biased toward the native configuration [27].

An example of an established MD approach was proposed by Sugita and Okamoto [26] which is a replica-exchange method for protein folding. The appeal of their approach was that it could overcome the multiple minima problem by exchanging non-interacting replicas of the system at several temperatures [26]. Their insight was to take random walks in *energy space* not *probability space* by avoiding the use of Boltzmann weighting.

Their methodology for the replica exchange method consists of M non-interacting replicas of the original system (of N atoms) in the canonical ensemble at different temperatures. The replicas are arranged such that there is always exactly one copy of the system at each temper-

ature and then there is a one-to-one correspondence between replica systems and temperatures [26].

However, even if their weighting is known *a priori*, they still need to determine the optimal temperature distribution [26]. Also the method, like any MD approach, is computationally demanding since it requires the simulation of many atomistic systems at a wide range of temperatures.

While computational power is on the increase there exists other regimes which are becoming more successful in protein structure prediction and folding, which are computationally cheaper. For example Monte Carlo methods are playing an increased role in these areas for which the explicit time dependence is not the ultimate goal [6].

## Protein Threading

Suppose we have a sequence  $s$  of known structure, can we determine the structure of a sequence  $s'$  that is homologous<sup>3</sup> to  $s$ ? The fact that  $s$  and  $s'$  are homologous could be derived from experimental biological data or alignment distances [25].

This is essentially the protein threading problem which relates to problem B in section 1.3. The basic idea is to use the known structure of  $s$  to guide the secondary and tertiary structure prediction for  $s'$ .

It is another optimization problem and was shown to be NP-complete by R.H.Lathrop in 1994 [28].

Through the use of experimental data or a protein data bank such as RCSB PDB<sup>4</sup> it is possible to construct a program which automatically searches this databank for homologous sequences to  $s'$  and then predict its structure to some degree of error. There are many programs and methods for doing this for example see [29] [30].

While this approach has its successes it is not completely blind, as it depends on currently known structures, and hence is in some sense scientifically incomplete. It is my opinion that it is ultimately more satisfying to find and understand the mechanisms of the system and then use this knowledge to make predictions.

## Other Algorithmic Approaches

The protein folding problem has attracted many computational approaches, some being very sophisticated. For example sequential importance sampling, PERM<sup>5</sup>[42] and other chain growth methods are in use in exploring the energy space of proteins in the HP model [36].

Also as the protein folding problem can be formed as a combinatorial optimization problem, ingenious and unexpected methods have come from the fields of mathematics and computer science [9]. Some of these include; genetic algorithms, ant colony models [46] and constraint-based algorithms.

---

<sup>3</sup>*Homologous*: having the same relation, relative position or structure

<sup>4</sup>[www.rcsb.org/pdb/home/home.do](http://www.rcsb.org/pdb/home/home.do)

<sup>5</sup>Pruned-enriched Rosenbluth method.

While these approaches, as with the protein threading paradigm, will advance our ability to predict the quaternary structure of proteins and help solve problem B (*see* 1.3), it is however not attacking the essence of the physical problem at hand. This physical problem being of a statistical mechanical nature.

### 1.5.1 The Work of Wust and Landau

A very successful regime for the study of the HP model was presented to the arXiv community in 2012 by Thomas Wust and David P. Landau [9]. Their work is mentioned here as it describes the only generic and fully blind Monte Carlo sampling scheme that can reproduce all known ground state energies and bettering one (for 3D103)[9]. Their scheme also allows the computation of thermodynamic and structural quantities at any temperature such as the specific heat capacity and the radius of gyration [9]. Their approach has also proven powerful for exploring the low-temperature behaviour of the self avoiding proteins even for  $N \gg 1000$  [11].

They use Wang Landau sampling described in section 3 and detailed in [7] to compute the density of states. Since the density of states does not depend on temperature in the canonical ensemble (*see* section 3.1) they were able to compute observables over the entire temperature range as shown in figure 7 [9].

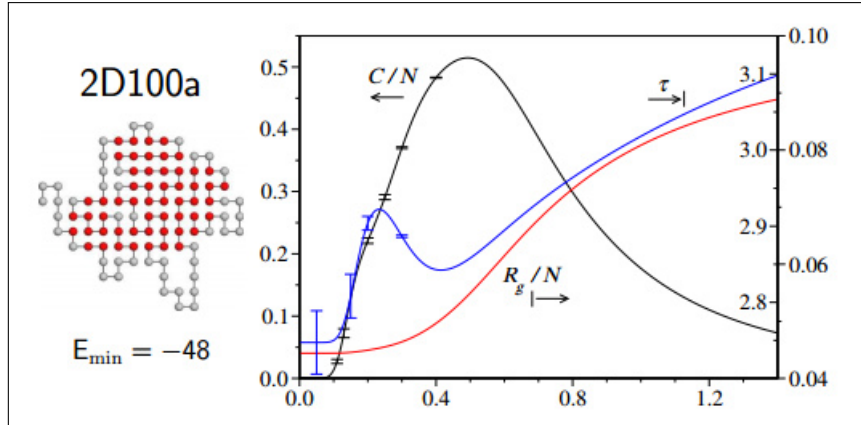


Figure 7: The computation of observables at different temperatures for sequence 2D100a. Specific heat capacity  $C/N$  is shown in black (left ordinates), root mean squared radius of gyration  $R_g/N$  is shown in red (outer right ordinates) and tortuosity  $\tau$  is shown in blue (inner right ordinates).

Their implementation is continuously compared to its close competitors, nPERMis<sup>6</sup> and FRESS<sup>7</sup>, and seems to have beaten them in the quality of the ground state energies and in computational efficiency. However direct comparison with other algorithmic methods briefly outlined here 1.5 was not done.

Their general paradigm, and the specific conclusion that implementation of trial move sets

<sup>6</sup>'New' Pruned Enriched Rosenbluth Method 'Importance Sampling'

<sup>7</sup>Fragment Re-growth via Energy-guided Sequential Sampling (FRESS)

is vital to a well-performing simulation of HP lattice proteins [9], seems a valid foundation to focus ones attention on in venturing into this rapidly growing area of biophysical simulation.

## **1.6 Aims of this project**

The general aim of this project is to investigate the behaviour of heteropolymer chains with protein-like sequences in a computational manner. More specifically to compute thermodynamic observables which will give insight into the folding/un-folding behaviour of protein chains. To achieve the native states of typical 2D benchmark sequences and make a clear comparison with other methods. Another aim is to achieve some sort of parallelism within the simulation, using a new or unknown scheme which has not been explicitly implemented to this particular problem. I would also like to obtain broad knowledge of this niche field in quantitative biology and understand the various approaches used to obtain understanding of protein folding.

## 2 Necessary Theory

### 2.1 Statistical Mechanics

#### 2.1.1 Canonical Ensemble

The protein chain, which consists of connected particles (monomers), can be assumed to exist in an aqueous solution which acts as a heat reservoir [4]. So that the protein chain is a small subsystem within the heat reservoir. Since the particles of the protein chain remain part of the subsystem we can use canonical ensemble theory to describe it statistically.

Therefore the protein chain is described by an ensemble of fixed temperature instead of fixed energy, since it exchanges energy with the solution around it.

Let the labels 1 and 2 denote the protein subsystem and the heat reservoir respectively. Working in the micro-canonical ensemble for the whole system, the total number of particles and energy are simply the sums of those in the system 1 and 2:

$$N = N_1 + N_2 \quad (2)$$

$$E = E_1 + E_2$$

where

$$N_2 \gg N_1 \quad (3)$$

$$E_2 \gg E_1$$

It is reasonable to assume that both systems are macroscopically large and that  $N_1$  and  $N_2$  remain fixed. The energies  $E_1$  and  $E_2$  fluctuate because the boundaries between the two subsystems allow energy exchange.

The goal is to find the phase-space density  $\rho_1(s_1)$  for system 1 in its own phase space. It is directly proportional to the probability of finding system 1 in the state  $s_1$  with no regard of the state of system 2. We see that it is proportional to the phase-space volume of system 2 in its own phase space with an energy  $E_2$ . Taking the proportionality constant = 1 we have:

$$\rho_1(s_1) = \Gamma_2(E_2) \equiv \Gamma_2(E - E_1) \quad (4)$$

Since equation 3 is assumed it is appropriate to Taylor expand the entropy of system 2 for small  $E_1$ :

$$k_B \cdot \ln[\Gamma_2(E - E_1)] \approx S_2(E) - \frac{E_1}{T} \quad (5)$$

where  $k_B$  is Boltzmann's constant and  $T$  is the temperature of system 2. In the thermodynamic limit for system 2 i.e.  $N_2 \rightarrow \infty$  the density function for system 1 becomes:

$$\rho_1(s_1) = \exp[S_2(E)/k_B] \cdot \exp[-E_1/(k_B T)] \quad (6)$$

The first factor in equation 6 is a constant and hence can be omitted after a normalization procedure. The energy of system 1,  $E_1$ , can be replaced by the Hamiltonian for the protein chain in the HP model using equation 1:

$$E_1 = H_1(s_1) \quad (7)$$

Therefore, omitting indices since system 2 is no longer relevant, the Boltzmann weight for a system at temperature  $T$  is

$$\rho(s) = \exp[-H(s)/(k_B T)] \quad (8)$$

which defines the *canonical ensemble*.

The partition function, with which all thermodynamic quantities can be derived from, is introduced as:

$$Z \equiv \sum_s \exp[-H(s)/(k_B T)] \quad (9)$$

where the summation is over all states. The partition function can also be written as:

$$Z \equiv \sum_E g(E) \cdot \exp[-E/(k_B T)] \quad (10)$$

where the sum runs over all energy values and  $g(E)$  is the density of states. Here  $E$  is the energy value of the protein chain computed via equation 1.

### 2.1.2 Energy Fluctuations and Observables

Let  $U$  be the mean internal energy of system 1 (the protein) which is given by the ensemble average of the Hamiltonian. Using equation 10 as the partition function  $U$  is

$$U = \frac{\sum_E E \cdot g(E) \cdot \exp[-E\beta]}{Z} \equiv \frac{\sum_E E \cdot g(E) \cdot \exp[-E\beta]}{\sum_E g(E) \cdot \exp[-E\beta]} \quad (11)$$

where  $\beta = \frac{1}{k_B T}$ . Differentiating  $U$  w.r.t  $\beta$  we obtain:

$$\frac{\partial U}{\partial \beta} = -\frac{\sum_E E^2 \cdot g(E) \cdot \exp[-E\beta]}{\sum_E g(E) \cdot \exp[-E\beta]} + \frac{(\sum_E E \cdot g(E) \cdot \exp[-E\beta])^2}{(\sum_E g(E) \cdot \exp[-E\beta])^2} \quad (12)$$

changing variables:

$$\frac{\partial U}{\partial \beta} = \frac{\partial U}{\partial T} \cdot \frac{\partial T}{\partial \beta} = -k_B \cdot T^2 \cdot \frac{\partial U}{\partial T} \quad (13)$$

where we recognise that the last partial derivative w.r.t  $T$  can be replaced with,  $C_V$ , the specific heat capacity.

So  $C_V$  is expressed as:

$$C_V = \left( \frac{\sum_E E^2 \cdot g(E) \cdot \exp[-E\beta]}{\sum_E g(E) \cdot \exp[-E\beta]} - \frac{(\sum_E E \cdot g(E) \cdot \exp[-E\beta])^2}{(\sum_E g(E) \cdot \exp[-E\beta])^2} \right) \cdot \frac{1}{k_B T^2} \quad (14)$$

The free energy  $F(T)$  is defined as:

$$F(T) = -k_B \cdot T \cdot \ln[Z] \quad (15)$$

where  $Z$  is the canonical partition function. Therefore to compute the free energy within the simulation one notes the more explicit form:

$$F(T) = -k_B \cdot T \cdot \ln\left[\sum_E g(E) \cdot \exp[-E\beta]\right] \quad (16)$$

Then the entropy,  $S(T)$  is then easily computed as:

$$S(T) = \frac{U(T) - F(T)}{T} \quad (17)$$

## 2.2 Probability Theory

### 2.2.1 Markov Chains

If we let the process that evolves the system be a stochastic one, so at discrete times  $t_1, t_2, t_3, \dots$ , the system is in a state  $W_t$  at time  $t$  which belongs to the set of all possible states denoted  $\{S\}$ . The conditional probability that  $X_{t_n} = S_{i_N}$  is given by:

$$P(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}, X_{t_{n-2}} = S_{i_{n-2}}, \dots, X_{t_1} = S_{i_1}) \quad (18)$$

assuming that the state of the system was, in the previous time, in state  $S_{i_{n-1}}$ . If the immediate state only depends on the preceding state i.e.

$$P(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}) \quad (19)$$

the stochastic process is then a Markov process and the set of states  $X_t$  is known as a Markov chain. Equation 19 is also referred to as the transition probability to go from one state to the next.

### 2.2.2 Non- Markovian Schemes

As explained in [10] Markov processes are the exception. Most stochastic systems and simulation models are intrinsically non- Markovian. A Markovian system is one where the distributional functions are solely given in 2.2.1, however in general one needs a different mathematical scheme to define the distribution of states.

Since Wang Landau sampling 3.1 has inherent history, through collected histograms and knowledge of old paths through energy space, it is a non- Markovian scheme (as noted in [9]).

### 2.2.3 Ergodic Process

In statistical theory a stochastic process is **ergodic** if its statistical properties can be deduced from a random sample of that process. The idea is that the random sampling of the process



meaningfully represents the average statistical properties of the entire process [50].

#### **2.2.4 Ergodic Hypothesis**

In computational physics it is more practical to view ergodicity as the ability to sample all of configuration space [6]. In the case of finite protein folding there does not exist the phenomenon of spontaneous symmetry breaking, so the entire phase space is reachable at all times. This means there will be no intrinsic ergodicity breaking.

In relation to simulations, it is of utmost importance that the operations which evolve the system can in principle take it through all of phase space in a finite amount of time. Since polymer dynamics require specialised and non-trivial move algorithms (see section 3.5) it is a danger that the simulation becomes non-ergodic and yields incorrect statistical results.

## 3 Methodology

### 3.1 Monte Carlo Methods

#### Reminder of Metropolis

Monte Carlo simulations are used extensively in science when the system at hand is sufficiently complex enough to be intractable analytically. The key to Monte Carlo simulation is to use sequences of random numbers to evolve the system or to sample integrals.

The workhorse of Monte Carlo simulations has been the Metropolis-Hastings importance sampling scheme, a good general review is given here [32] and for applications for statistical physics here [6].

The Metropolis scheme can be briefly stated as follows:

---

#### METROPOLIS SCHEME

1. Choose an initial state of the chain.
  2. Propose a trial move selected at random from the set.
  3. Compute the energy change  $\delta E$  which results from the conformational change.
  4. Generate a random number  $ran$  where  $0 < ran < 1$ .
  5. If  $ran < \exp[-\delta E \cdot \beta]$  accept the move.
  6. Go to step 2 and repeat  $n$  times.
- 

#### 3.1.1 Wang Landau Sampling

Contrary to the Metropolis Hastings scheme in which the acceptance criterion is based on the difference in energy via Boltzmann weighting, Wang- Landau sampling has its acceptance criterion based on the inverse of the density of states [7].

Say, for example, a protein chain in configuration  $a$  has some energy  $E_a$  computed using equation 1. If we make a move on the chain i.e. perturb its configuration such that it now has energy  $E_b$  where the configuration has gone from  $a \rightarrow b$ . Moves are accepted according to the probability:

$$p(E_a \rightarrow E_b) = \min\left(\frac{g(E_a)}{g(E_b)}, 1\right) \quad (20)$$

We want to ultimately compute the canonical partition function as shown in equation 10, which entails approximating the DOS  $g(E)$ . For equation 20 to work, we start with a simple guess of the DOS at each discrete energy level. This is because  $g(E)$  is not known a priori but

it is possible to iteratively refine the initial guess such that it converges to the correct DOS for the system.

Let the initial guess be simple i.e.  $g_0(E) = 1$  for all  $E_1, \dots, E_n$ . Then following each move, whether accepted or rejected, we update the DOS for the resultant energy level  $E$  via:

$$g(E) \rightarrow g(E) \cdot f_i \quad (21)$$

The modification factor,  $f$ , is also modified according to a flatness criterion for the collected histogram of energies. The factor starts out as  $3 > f_0 > 1$ <sup>8</sup> and if the histogram is flat, up to some pre-determined standard,  $f$  is reduced:  $f_{n+1} = (f_n)^{\frac{1}{2}}$ . The histogram entries are then reset to zero and the process begins again but with a reduced modification factor.

The aim is to have  $\lim_{f \rightarrow 1} g_{approx}(E) = g_{exact}(E)$ . Since this limit converges it is appropriate to foster an accuracy cut-off for the modification factor. This can be chosen to be  $f_{final} \approx e^{10^{-8}}$  [9].

In this simulation the DOS spans many orders of magnitude and hence may lead to numeric overflow in the 'long double' data type in C/C++ (as happened during initial stages). This leads to '-nan' for the thermodynamic observables. It is preferable to work with the natural logarithm of the DOS where initially  $\log[g_0(E)] = 0$  and the update procedure is then:

$$\log[g(E)] \rightarrow \log[g(E) \cdot f] \equiv \log[g(E)] + \log[f] \quad (22)$$

and it is still reasonable to keep reducing  $f$  directly.

The detailed step-by-step Wang-Landau scheme for this simulation is as follows:

---

### WANG-LANDAU SCHEME

1. Set a pre-defined range of discrete energies (not too large to be cumbersome) that the protein may take.
2. Initialise:  $X(E_i) = 0$ ,  $H(E_i) = 0$  and  $F = 1$  (where  $X(E)$  and  $F$  represents  $\log[g(E)]$  and  $\log[f]$  respectively).
3. Initialise the chain positions.
4. Perform a random move but remember to store the previous energy and positions.
5. Compute  $\eta = \exp[X(E_1) - X(E_2)]$  and generate a random  $\# = ran$  between 0 and 1.
6. **IF** ( $\eta > ran$ ) accept the move **ELSE** return to the old configuration.
7. Update the Histogram and the DOS:  $H(E)_{n+1} = H(E)_n + 1$ ,  $X(E)_{n+1} = X(E)_n + F$ .
8. **IF**  $H(E_i) > q \cdot \langle H(E) \rangle$  for all visited energies **DO**  $F_{n+1} = \frac{F_n}{2}$ . Reset the histogram.

---

<sup>8</sup>In the literature (see [9] and [12]) normally  $f_0 = e^1$  however there is no systematic way to determine the most efficient starting modification factor.

9. **ELSE** Go to step 4.
10. Repeat until  $f \approx 10^{-8}$  or after a certain amount of time  $t$ .
11. Compute thermodynamic observables using  $\log[g(E)]$  etc.

---

For step 8 a flat histogram occurs when the histogram value in each energy bin is above  $q \cdot \langle H(E) \rangle$ . The parameter  $q$  can be set to any value  $< 1$ , although, as will be discussed later, the precision of the histogram directly affects WL convergence and will have to depend on the chain length.

For new protein sequences, where the energy minimum is not known, and for existing sequences the Wang Landau scheme requires an energy range to sample the DOS from. Since this energy range is not known a priori it seems useful to conduct a pre-WL-run to ascertain the energy ranges. This is a time consuming procedure because many low energies are only visited during the final stages of the simulation.

It seems more viable, also retaining the blindness of the approach, to only have the DOS and histogram updated for visited energy sites. So the Wang Landau algorithm and code is modified so that it checks whether the energy has been visited before. In this work another array called 'visited[ENERGY]' is initialised to zero at the beginning of the simulation and once the energy is visited its corresponding array value will be set equal to 1. This value will remain = 1 for the rest of the Monte Carlo iterations. Once a new energy has been found the histogram (not the modification factor) is reset to zero.

To accompany this, the flatness checking of the histogram occurs every  $10^6$  iterations so that the modification factor isn't updated too prematurely for few visited energies.

This will provide a quicker and easier way to attain the energy range without performing previous simulations.

It is also worth emphasizing that the DOS and histogram of a resulting configuration, which occurred due to a rejection of a proposed one, must be updated accordingly to ensure correct sampling of phase space. Not doing so would result in an incorrect estimate of the density of states and hence any observables derived from it would be devoid of physical meaning.

### 3.1.2 1/t algorithm

It has been shown and argued that the WL procedure presented above does not converge asymptotically to the correct density of states of the system [33] [34]. This is due to the saturation in the modification factor which occurs for high MC iterations, the cause of the saturation is due to the function which reduces the modification factor.

The cure for this which is presented in [33] is to have the reduction of the modification factor take on a functionality which depends on the MC time,  $t$ , only if all the relevant states of the system have been visited and that the modification factor is smaller than the current MC time.

Using the Ising model Belardinelli and Pereyra defined the MC time to be  $t = \frac{j}{N}$ , where  $j$

is the number of iterations attempted and  $N$  is the number of energy states available to the system. Following in a similar fashion the MC time in this simulation is defined as  $t = \frac{M}{\delta}$  where  $M$  is the number of iterations attempted and  $\delta$  is the energy range of the WL sampling scheme.

The step by step algorithm which alters the WL algorithm in the previous subsection is as follows:

---

### 1/t Scheme

1. Set a pre-defined range of discrete energies (not too large to be cumbersome) that the protein may take.
  2. Initialise:  $X(E_i) = 0$ ,  $H(E_i) = 0$  and  $F = 1$  (where  $X(E)$  and  $F$  represents  $\log[g(E)]$  and  $\log[f]$  respectively).
  3. Initialise the chain positions.
  4. Perform a random move but remember to store the previous energy and positions.
  5. Compute  $\eta = \exp[X(E_1) - X(E_2)]$  and generate a random  $\# = ran$  between 0 and 1.
  6. **IF** ( $\eta > ran$ ) accept the move **ELSE** return to the old configuration.
  7. Update the Histogram and the DOS:  $H(E)_{n+1} = H(E)_n + 1$ ,  $X(E)_{n+1} = X(E)_n + F$ .
  8. After some fixed sweeps (100000 iterations in this case), if, for all  $E$ ,  $H(E) \neq 0$  then  $F_{n+1} = \frac{F_n}{2}$ . Reset the histogram.
  9. **IF**  $F_{n+1} \leq t^{-1}$  **DO**  $F_{n+1} = t^{-1}$  and in what follows  $F$  is updated at each MC time for the rest of the simulation run (Step 8 is no longer used).
  10. Stop the simulation after a fixed elapsed time or until the modification factor is small enough to warrant convergence.
  11. Compute thermodynamic observables.
- 

### Problems In Implementation

This scheme, while seemingly optimal in abstraction, is difficult to implement. Runs were performed for sequences with  $N < 20$  and many converged via this scheme however for benchmark sequences convergence via (t) functionality is almost impossible under the current definition of MC time. This is due to the slower rate of modification factor reduction which occurs as  $N \rightarrow \infty$ .

It is difficult to consider what changes can be made to the MC time without making it too sequence and simulation dependent (non-blind). This is noted in [9] where the tweaking

procedures to ensure this scheme works may be too costly in time and effort. However WL sampling still estimates the DOS well if the histogram flatness criterion and final modification factor threshold are stringent enough.

This scheme is still embedded within the code in case the convergence rate increases, however to make this algorithm live up to its potential requires dedicated testing and tweaking of code.

### 3.1.3 Detailed Balance

WL sampling is a non- Markovian process <sup>9</sup> where it has been shown to provide a valid estimation of the density of states [48] [49] without depending on detailed balance. However it is still vital that the trial moves respect detailed balance to avoid troubling systematic errors [9]. I ensure detailed balance by choosing trial moves at random but with constant probability. Since detailed balance is guaranteed if a trial move is reversible and the reverse/ original move have the same probability.

In every trial move if there is a choice to go in multiple directions they are chosen with equal probability. The only preference of 'choice' are the trial move ratios which are fixed throughout a run of the simulation.

As the modification factor,  $f$ , converges to 1 detailed balance is recovered since:

$$\frac{1}{g(E_1)} \cdot \pi(E_1 \rightarrow E_2) = \frac{1}{g(E_2)} \cdot \pi(E_2 \rightarrow E_1) \quad (23)$$

---

<sup>9</sup>(see 2.2.2 for description)

### 3.2 Lattice System

A protein chain, in this implementation, exists on a square 2D lattice of length  $L$  where monomers can be located via column,  $i$ , and row,  $j$ , coordinates stored as  $\epsilon_{ij}$  (see equation 24).

$$\epsilon_{ij} = (i, j) \quad (24)$$

It is computationally cheaper to work using a 1D array which maps onto the 2D lattice. Let  $\epsilon \in \mathbb{Z}$  be an element of such an array and impose that  $\epsilon \in \{0, \dots, L^2 - 1\}$ . For example a 2D lattice in which  $L = 4$  is pictured in table 2.

|   | 0  | 1  | 2  | 3  |
|---|----|----|----|----|
| 0 | 0  | 1  | 2  | 3  |
| 1 | 4  | 5  | 6  | 7  |
| 2 | 8  | 9  | 10 | 11 |
| 3 | 12 | 13 | 14 | 15 |

Table 2: An example 2D lattice with locations stored in a 1D array.

One can retrieve the row and column values from any  $\epsilon$  using:

$$i = \epsilon \bmod L \quad (25)$$

$$j = \frac{\epsilon}{L} \quad (26)$$

where in equation 26 the value is rounded down to the nearest integer.

It will be conducive to outline the relationships between neighbouring lattice sites and to define the values of  $\epsilon$  which form the boundary.

The element,  $\epsilon'$ , directly above a given  $\epsilon$  is given by  $\epsilon' = \epsilon - L$  and the element directly below is given by  $\epsilon' = \epsilon + L$ . The element directly to the right of a given  $\epsilon$  is given by  $\epsilon' = \epsilon + 1$  and to the left is given by  $\epsilon' = \epsilon - 1$ .

The values of  $\epsilon$  which lie on the upper boundary satisfy:  $\epsilon < L$ . The values of  $\epsilon$  which lie on the lower boundary satisfy:  $L(L - 1) \leq \epsilon \leq L^2 - 1$ .

The values of  $\epsilon$  which lie on the right boundary satisfy:  $\epsilon = aL - 1$  for  $a \in \{1, 2, \dots, L\}$ . The values of  $\epsilon$  which lie on the left hand boundary satisfy:  $\epsilon = bL$  for  $b \in \{0, 1, \dots, L - 1\}$ .

An example of how the amino acid residues would be placed onto the 2D lattice with location values stored in a 1D array is shown in table 3. This description is, while trivial, essential to the programming of the simulation as all operations on a chain are essentially operations on this lattice system.

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | • |   |   |   |
| 1 | • | • |   |   |
| 2 |   | • | • |   |
| 3 |   |   |   |   |

Table 3: An example of (H) residues on a 2D lattice of side length  $L = 4$ .

### 3.3 Dynamical Trapping

Recently (2015) it has been shown that Wang Landau sampling of continuous systems suffers from a phenomenon known as dynamical trapping [44]. The trapping is when the WL sampler only updates the same density of states and histogram for many iterations. The trapping is caused by the random walker coming close to extrema on the energy landscape and should be distinguished from the critical slowing down in conventional MD or MC simulations [44].

The works mentioned in [44] were all simulations of physical systems using continuous degrees of freedom. The problem of dynamical trapping can still be an issue for discrete models, as is used here, with rough energy landscapes. The compact configurations of proteins near the native region will increase the rejection rates of most moves within the trial set (see section 3.5), and whilst the FRW move does have the ability to escape these tight configurations (see section 3.5.1) it may not be the optimal solution on its own. Whilst the simulation is rejecting most local moves and some non-local moves on monomers that are completely surrounded by others, it will create spikes in the DOS and histogram (example shown in figure 8) which will greatly damage the accuracy of computed observables and convergence time. Also when trapped, the random walker misses entire or even several stages of Wang-Landau modification factor reduction, which leads to inadequate sampling of conformational space and a rough estimate of the DOS even if the modification factor is reduced to very small values [44].

To prevent dynamical trapping from occurring *Koh, Sim and Lee* proposed a simple parallel trajectory-exchange scheme [44]. This scheme consists of running multiple WL samplers for the system at hand and randomly swapping configurations with each other at regular intervals of MC time. This method is different to that proposed by *Vogel, Li, Wust and Landau* [45](Replica-Exchange Wang Landau) which proposes the exchange of configurations existing within overlapping energy windows.

Each WL sampler in the trajectory-exchange scheme has its own private estimation of the DOS and thermodynamic observables, the scheme merely imposes the regular swapping of configurations. The main mechanics of the idea can be understood effectively through figure 9.

In [44] they surmised that  $T < 1000$  (where  $T$  is the swapping period). In this work  $N_P$ , the number of processes/threads, is large enough to warrant the use of Gaussian statistics.

The most natural language to implement this scheme, in my opinion, is via MPI <sup>10</sup>(Message

---

<sup>10</sup>For details on the MPI language and inherent library routines: 'Gropp, William., Lusk, Ewing. and Skjellum,



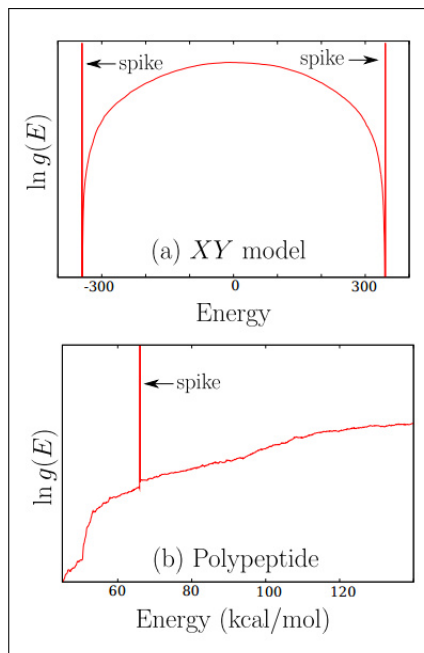


Figure 8: Spiked DOS for (a) the frustrated XY model and (b) the 8-mer poly-alanine. For details see [44].

Passing Interface) since it is very simple to adapt the serial code in order to impose trajectory swapping. In this work the master process produces a random source ID for every process ID such that the source process sends its trajectory to the destination process. In this way every process has its trajectory swapped in a random manner.

Once the WL sampling routine is complete each process then computes thermodynamic quantities as outlined in section 2.1.2, the results are then averaged and statistical error analysis is then conducted.

A problem arises: If one process attains the native or a near native state (which is very compact), swapping the trajectories will not make it more likely to escape this configuration due to every process having the same trial move set ratios. So this could cause the downfall of the trajectory swapping method. Very low temperature configurations will eventually loosen since not every move will be rejected, but as this occurs using many processes they will not show extreme spikes in the DOS. The configuration will hop between processes whilst gradually unwinding. As long as swapping is very regular this problem does not pose any threat.

Also the fact that lower configurations may be shared by many processes before being completely changed helps each process explore the low temperature regions of phase space.

Whilst the REWL (overlapping energy windows) scheme has been implemented successfully for a more sophisticated variant of the HP model [47], it is easier to augment existing serial code into a parallel framework using simple trajectory swapping whilst still making major

---

Anthony. *Using MPI: Portable Parallel Programming with the Message-Passing Interface (2nd Edition)* is recommended by the author.

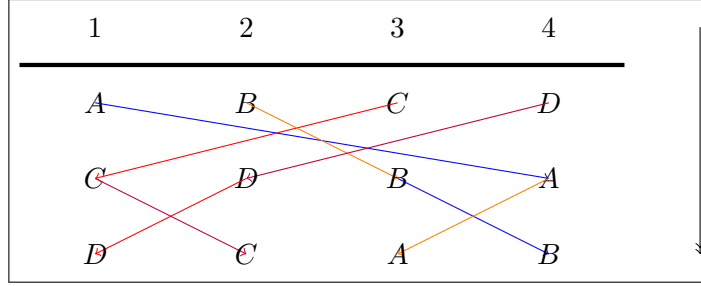


Figure 9: The numbers represent the thread ID's and letters represent arbitrary configurations. Trajectories are swapped through random shuffling. The arrow to the right represents the direction of MC time.

improvements to the robustness of the simulation. Hence incorporating the trajectory-exchange parallel scheme allows for a simple and efficient way to explore the thermodynamics of the HP model.

### 3.4 Logistics of the parallel implementation

After the threading environment has been created the root process reads in the (H)(P) sequence from a file and assigns the values to an array `HP[ ]`, then it initialises the `visited[ENERGY]` array to 0. The size of the visited array, and any array which has an argument of ENERGY, can be set to the length of the protein chain for safety. The `HP[ ]` and initialised `visited[ ]` arrays are then sent to all processes.

Each process initialises the chain in the same manner but is assigned a personal seed number,  $S$ . A 'master' seed,  $S_M$ , is chosen from an external random number generator and the seed for each process is generated via:

$$S = (S_M + myid) \cdot a \cdot (myid + b) \quad (27)$$

where  $myid$  is the process id and  $a$  and  $b$  are arbitrary positive integers.

The snippet of code which implements the trajectory swapping is shown in appendix C. A random source process is chosen to send its trajectory to a destination process. The destination process goes from 0 to  $numprocs - 1$  so that each process has a new configuration.

The minimum energy from all simulations were found via a `MPI Reduce()` function. Each process then computes thermodynamic quantities in a pre-defined temperature range. Statistical analysis was then conducted externally.

### 3.5 Trial Move Implementation

In polymer and past HP model simulations there are tried and tested trial move sets which respect the LCSAW condition. These include the end-bond flip (figure 10), kink flip (figure 11) and the crankshaft move (not used in this simulation). A trial set consisting of these moves only, does not respect the ergodicity condition: that all of configuration space is reachable. However the inclusion of pull moves and pivot moves restores ergodicity [9] [12].

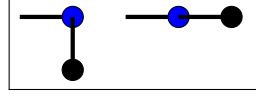


Figure 10: An example of an end-bond flip move where the penultimate monomer or second monomer acts as an axis (blue) so that the 1st or last monomer can rotate about it.

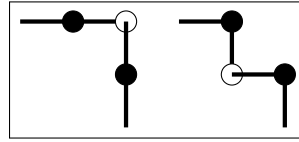


Figure 11: An example of a kink flip move where the white monomer is at a corner between its sequential neighbours and 'flips' to the opposite corner if it is free.

In this simulation the trial move set consists of pull, kink flip, pivot, bond rebridging and fragment random walk moves. As one shall see this trial move set necessitates the inclusion of the end-bond flip move. This move set is different to that used in [9] and [12], in the fact that here the fragment random walk move is introduced and all moves in this simulation are coded originally and may be implemented slightly differently.

*Pull Move:* A monomer is chosen at random to act as the *primary monomer*, this means it is the first monomer to move to a free neighbouring position. This future position of the primary monomer is determined by the *anchor monomer*, where it will move to its right or left or above or below it depending on the availability of these positions on the lattice. If the monomer is at the end or start of the chain it can only have the penultimate monomer or second monomer as the anchor monomer respectively. If the monomer has a sequence value  $s$  such that  $1 < s < N$ , where  $N$  is the total number of monomers, then the anchor monomer  $s_a$  is chosen, with equal probability, between  $s_a = s - 1$  and  $s_a = s + 1$ . Once the primary monomer has moved to a suitable position next to the anchor monomer, the secondary monomer (next to primary monomer on the sequence) slots into a suitable position which keeps it connected to the primary monomer. The rest of the chain 'slithers' along occupying the positions of relevant old monomers which ensures the LCSAW condition is fulfilled (figure 12).<sup>11</sup>

<sup>11</sup>N.B. The original pull move consisted of pulling the rest of the chain along every time, which while still effective was unnecessary and potentially unrealistic. This move algorithm was modified so that it stops once it respected the SAW condition, which makes it affect less monomers on average.

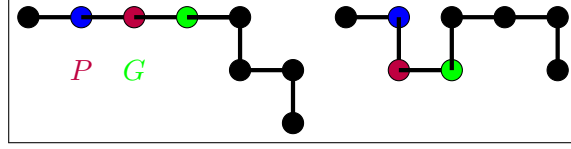


Figure 12: An example of a pull move where the anchor monomer (blue) remains fixed and the primary monomer (purple) will move to  $P$  and the secondary monomer (green) moves to  $G$ . The rest of the chain 'slithers' behind to keep the sequence and length of the chain fixed.

*Pivot move:* This move starts by choosing a monomer at random with sequence value  $1 < s < N$  to act as another anchor monomer so that it acts as an axis in which another part of the chain rotates about it. It makes no difference to the configuration of the protein chain if a rotation is executed around the 1st monomer or  $N$ th monomer since this doesn't change the internal structure and hence will remain stationary in energy space. So these rotations are omitted for convenience in this simulation. So when a random monomer has been chosen an acceptable move is to either rotate the part of the chain with monomers having sequence values  $< s$  or monomers with sequence values  $> s$ . The algorithm chooses either case with probability  $\frac{1}{2}$  to ensure that no biases occur. Also rotations either anticlockwise,  $\curvearrowright$ , or clockwise,  $\curvearrowleft$ , are acceptable and hence the algorithm decides to undertake such rotations with probability  $\frac{1}{2}$ . The rotations then occur leaving the rotated structure internally invariant but it's relationship with the rest of the chain will change. The pivot algorithm always checks whether the future space of the monomers are available, otherwise the move is rejected. An example of a pivot move is shown in figure 13.

The pivot move was included to accelerate convergence in the DOS computation via WL sampling [9], also it ensures that the entire phase space of the system is attainable.

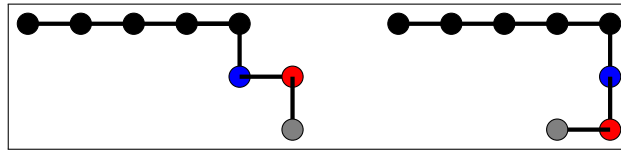


Figure 13: An example of a  $\curvearrowleft$  pivot move where the anchor monomer (blue) remains fixed and the red and grey monomers move according to the change in directionality relative to the preceding monomer. Initially red was to the right of blue and now it is below it also grey was below red and now it is to the left of it.

The pivot algorithm proposes the future positions of monomers on the 'to be' rotated structure via operations on the *directionality*. The *directionality* can be defined as the relative direction that a monomer  $B$  is relative to a monomer  $A$ . A table showing how anticlockwise and clockwise rotations affect the directionality is shown in table 4.

Directionality can be stored as an integer quantity  $d \in \{1, 2, 3, 4\}$  in 2D where  $\rightarrow = 1$ ,  $\leftarrow = 2$ ,  $\downarrow = 3$  and  $\uparrow = 4$ . The routine *buddycheck*(*int*  $N$ , *int* 'position of monomer  $A$ ', *int* 'position



|               |  |  |
|---------------|---|---|
| $\rightarrow$ | $\uparrow$  | $\downarrow$  |
| $\downarrow$  | $\rightarrow$   | $\leftarrow$  |
| $\leftarrow$  | $\downarrow$  | $\uparrow$  |
| $\uparrow$    | $\leftarrow$  | $\rightarrow$   |

Table 4: How relative direction is changed under the two rotations.

of monomer  $B'$ ) (ref appendix of buddycheck) returns  $d$  as the directionality of monomer  $B$  to monomer  $A$ . Once the operation on the directionality has occurred successfully and the future positions are indeed available, the pivot move executes a move.

*Kink flip move:* As seen in figure 11 the kink flip move only affects a monomer at a corner. There are 4 possible scenarios which allow a kink flip move to be performed shown in figure 14.



Figure 14: The four possible scenarios for a kink flip move, the orange  $O$  represents the future position of the primary monomer (orange). From left to right the names of the moves are as follows: *bottom left quadrant move*, *top right quadrant move*, *top left quadrant move* and *bottom right quadrant move*.

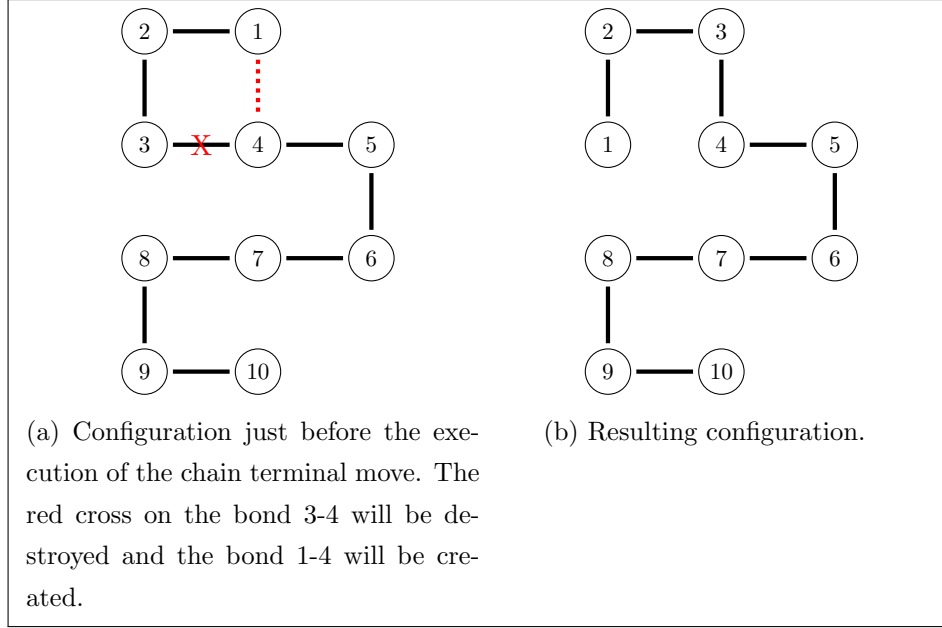
The `kinkflip(int  $N$ , int  $L$  (lattice side length), int  $L^2 - 1$ )` routine (insert ref to appendix for kinkflip code) searches for kinks in the chain beginning at  $s_i = 2$  and ending at  $s_f = N - 1$ . If there is a kink it will execute one of the four possible moves depending on whether the relevant future position is available or not. If a move gets rejected it continues along the chain looking for more kinks, this helps keep the rejection rate at a minimum. If any move is executed properly the routine closes.

### Bond Re-bridging

These moves do not change the position of the chain on the lattice, i.e. the array positions remain constant, but changes a number of the bonds of the chain and then re uploads the sequence onto it as to dramatically change the configuration. This move becomes useful in sampling low temperature phase space where compact configurations lead to high rejection rates for local moves like pull, pivot and kink flip.

There are two types of bond re-bridging moves used in this work namely chain-terminal and type II (see [35] for an in depth discussion).

*Chain Terminal:* This move consists of destroying a bond between monomers and then recreating a bond with a topological neighbour of  $N$  and  $1$  respectively. The destruction of a



bond must only occur between the topological neighbour of N and 1 and a connected neighbour on the sequence with dependence on whether N or 1 has been chosen. For example if 1 was chosen then its topological neighbour say, m, can only destroy its bond with m-1 on the sequence. If N was chosen then its topological neighbour, m, can only destroy its bond with m+1 on the sequence.

The chain terminal algorithm implemented here first chooses (with 50% chance) monomer 1 or N and then searches for a topological neighbour for which it can form a new sequence bond. After this search for topological neighbours the step in the procedure is pictorially shown in 15a.

Since positions of the monomers are stored in a 1-dimensional array space (see section 3.2) the chain terminal algorithm simply swaps the position of the connected neighbour, who is connected topologically to 1 or N, with 1 or N.

For example as with the before and after in figures 15a and 15b respectively the position of 3 is swapped with the position of 1. The rest of the chain is unaffected. In general, for the '1' case, the algorithm is outlined as:

1. Call topological neighbour m and connected neighbour m-1.
2.  $POS[1]=OLDPOS[m-1]$
3. FOR( $i=2; j=m-2; i < m-2; j > 1; i++; j--$ )  
 $[POS[i]=OLDPOS[j];]$

The steps are similar for the 'N' case.

Type II Move: This move is not restricted to the ends of the chain and destroys and recreates 2 bonds in contrast to only 1 in the chain terminal move. We define the 'quad' as the sub square which contains the monomers that dictate the implementation of the move and hence swapping of position values. An example of a 'quad' can be seen in figure 15.

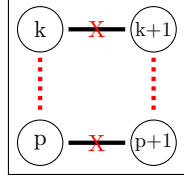


Figure 15: The red crosses represent bonds that will be destroyed and dotted lines represent future connected bonds.  $k$  and  $p$  are monomer values within the set  $\{1, \dots, N\}$  (Ignoring other monomers for clarity).

Using figure 15 as a reference, we note that the monomer numbers from  $k$  and  $p$  increase in the same direction. This creates a linear topology where, if we cement the new proposed bonds, no part of the sequence will ever be cut off. This means it will obey the LCSAW condition.

The pseudo algorithm, used here, for the type II move is as follows:

1. Select a monomer,  $p$ , at random.
2. Choose, at random, a connected neighbour,  $j$ , of  $p$  i.e.  $p-1$  or  $p+1$  (unless  $p = 1$  or  $N$ ).
3. Look for topological neighbours of  $p$  and  $j$  with the same relative direction (see figure 15 for reference).
4. IF (they form a quad)
  - DO step 5
  - else go back to step 1.
5. Find the smallest and largest monomer number in the quad set.
6. Impose constant positions:
  - $\text{POS}[\text{smallest}] = \text{OLDPOS}[\text{smallest}]$
  - $\text{POS}[\text{largest}] = \text{OLDPOS}[\text{largest}]$
  - $\text{POS}[1] = \text{OLDPOS}[1]$
  - $\text{POS}[N] = \text{OLDPOS}[N]$ .
7. Re-upload the other monomers correctly:
  - FOR( $i = \text{smallest} + 1; h = \text{largest} - 1; i \leq \text{largest} - 1; h \geq \text{smallest} + 1; i++; h--$ )
  - ( $\text{POS}[i] = \text{POS}[h];$ )

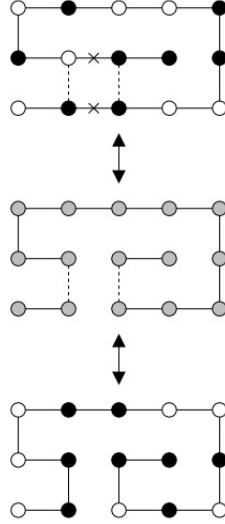


Figure 16: An example of a type II bond-rebridging move on a small lattice protein chain. Note the re-uploading of the HP sequence.

### 3.5.1 Fragment Random Walk

It is vital that the trial move set allows rapid coverage of configuration space but still gather a detailed picture of the rough energy and conformational landscape. This is to ensure that the DOS can be estimated quickly and accurately. Local moves such as pull, kink-flip and in some cases pivot moves only displace a relatively small amount of monomers which allows the Wang-Landau sampling scheme to gather detailed information. To aid with pivot moves (in cases where a large number of monomers are displaced) in accelerating global conformational changes [9], I have supplemented the trial set with the *fragment random walk* (FRW) move.

This move has not been implemented in [9], [12] or any other work since it has been invented here. Hence the inclusion of this move makes the trial move set used here unique.

The FRW move is partly inspired by FRESS (fragment regrowth Monte Carlo) [36] where an **internal** segment of the protein chain (of chosen length) is chosen at random and a new fragment is 'regrown' to replace it, hence causing a conformational change. The move used in FRESS is illustrated in figure 17. In FRESS a fragment regrowth move is only accepted if it obeys the Metropolis - Hastings criterion *see section 3* [36].

FRW is different to the regrowth of fragments used in [36] in that the fragments are not internal and are not necessarily of fixed size. Internal is defined as: the fragment having two fixed points that are monomers  $\in \{2, N - 1\}$ , so the FRW has only one fixed point in the same set of monomers.

The pseudo algorithm for the FRW move is as follows:

1. Pick a random monomer  $m \in \{2, N - 1\}$ .
2. With 50% probability monomers with sequence number  $n > m$  or  $n < m$  are chosen to



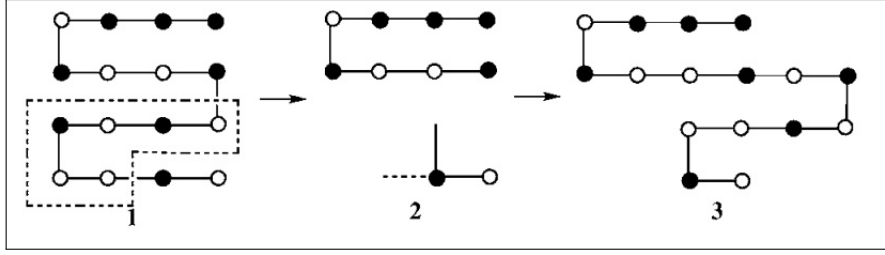


Figure 17: 1) The initial configuration with a randomly selected fragment. 2) Fragments are being produced at random, any fragments which do not connect to the fixed points are rejected. 3) A successfully regrown fragment and its resulting configuration. (*Picture originally published in [36] and gratitude goes to Zhang, Kou and Liu.*)

form the fragment.

3. Start the self avoiding random walk for the fragment.
4. If the walk violates the LCSAW condition and not all local positions have been tried, then try another position.
5. If all local positions have been exhausted then return to old configuration.
6. Else if the fragment random walk has been completed successfully exit move algorithm

An example of a successful FRW move is shown in figure 18.

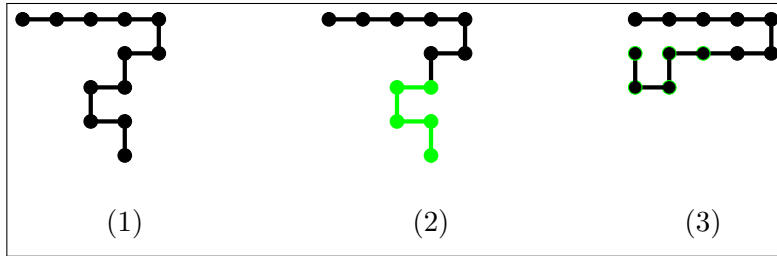


Figure 18: 1) The initial configuration before the FRW move. 2) A fragment is chosen. 3) The resulting configuration of a successful FRW move.

Why wasn't the move used in FRESS simply utilized in this scheme? The FRESS move, while having its own advantages, will suffer from high rejection ratios in low temperature conformations and does not rapidly change the energy as much as the FRW move. This is due to the fact that FRW has no limit on the fragment size meaning a large portion of the chain can be rapidly changed. Also the constraint of having two internal fixed points also will mean less acceptance and hence a slower acceleration of global conformational change, which is what the main purpose of 'non- local' move of this nature will be used for here.

So the potential advantages of the FRESS move, for example in possibly aiding the bond

re-bridging move in accessing low temperature configurations, does not compete with the advantages of the FRW move in rapidly changing the configuration of the chain.

### 3.5.2 LCSAW and Excluded Volume Barriers

As highlighted in section 1.4 valid configurations of the protein chain are those which respect the conditions that only one monomer can occupy a lattice site and that the chain is simply connected. It is absolutely essential to the Wang Landau sampling scheme and native state search that only valid configurations of the system are taken into consideration, since the resulting density of states will be wrong for the assumed system. Hence barriers which block any illegal configurations from entering the Wang Landau scheme have been imposed, if such an illegal configuration is found it is rejected and the last valid configuration becomes the present one. Once the old configuration has to be used again its corresponding histogram and density of states is updated accordingly.

### 3.5.3 Trial Move Testing

To ensure the trial move algorithms we operating as intended and were implemented in a somewhat random way, tests on each trial move algorithm were conducted. The tests involved running the move algorithms on their own (except the kink flip algorithm <sup>12</sup>) and manually checking the coordinates of the monomers after each move.

The random number generator used in this testing procedure and throughout this simulation is outlined in random.C (ref appendix).

Some brief example chain pathway diagrams, which show the configuration of the chain in increments of move time, are presented for the move algorithms. I programmed the test so that the user screen prints out the 1D array values  $\epsilon_i$  for  $i \in \{1, \dots, N\}$  which I then drew out the corresponding chain diagram.

### Pivot Move Tests

These chain pathways (figure 19) represent pivot move operations only on a 6-mer (HHPHHP), with 29 total attempt move operations and with random seed # 9062. Lattice side length  $L = 300$ . The configurations shown are the ones that actually changed the configuration as many were rejected. The acceptance ratio, for this test, was 5/29.

One can see that for the chain pathways in figure 19 in 5/29 successful moves the pivot move algorithm on its own has found 2 native degenerate states for the 6-mer.

A more thorough test was conducted which comprised of 500 moves and only the starting and ending configuration was recorded to check the chain was still intact. The test was on a 10-mer (HHPHHPHPPHH) using a random seed # 9062. The lattice side length  $L = 300$ . The

---

<sup>12</sup>This is due to the fact the chain started out as 'linear' where there were no kinks in the chain. To create new kinks the pullmove or pivotmove needed to be present and the kink move would act on any existing kinks. This allowed me to see if it actually worked.

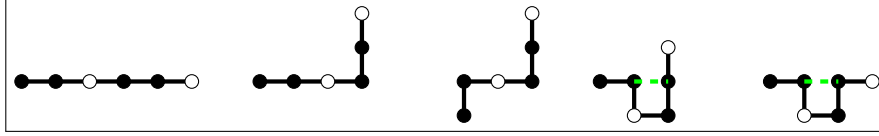


Figure 19: The evolution of this chain goes from left to right. (H) monomers are represented as black circles and (P) monomers are represented as white circles. The green dashed lines represent H-H contacts and for this chain represent native states, since the maximum number of H-H contacts is 1.

pictorial results of the test are shown in figure 20.

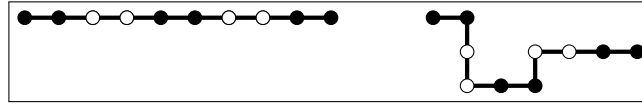


Figure 20: The 10-mer before any moves (left) and after 500 pivotmoves (right). The HP sequence (HHPPHHPPHH) of monomers remain invariant and the chain remains intact which means the moves respect the conditions of the HP model and LCSAW.

### Pull Move Tests

Using a 6-mer (HHPHHP) a sequence of chain pathways was produced using pull moves only, with 29 total attempt move operations and with a random seed # 9062. Lattice side length  $L = 300$ . The configurations shown in figure 21 are the first 5 configurations from the sequence (for illustration purposes) as all pull moves were successfully done.

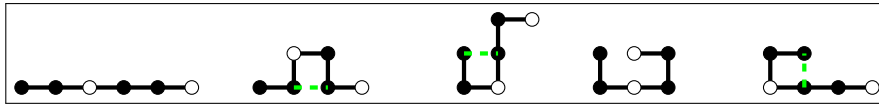


Figure 21: The first five configurations representing the evolution of the 6-mer via pull moves. One can see that three unique native states have been found.

For the pull move algorithm, as was done with the pivot move algorithm, a test was performed consisting of 500 moves in which only the starting and ending configuration was recorded. The test was on a 10-mer (HHPPHHPPHH) using a random seed # 9062 and with the usual lattice side length  $L = 300$ . The pictorial results of the test are shown in figure 22.

### Kink Flip Move Tests

The kink flip move was described in section 3.5. Since, in these tests, the chain starts as a linear one where no kinks are present it was necessary to include another move to create kinks to see if the kink flip move was functioning correctly. This does not affect the quality of the

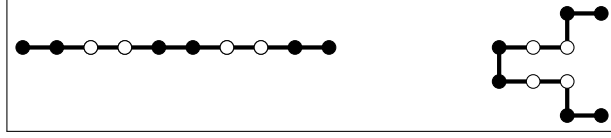


Figure 22: The 10-mer before any moves (left) and after 500 pull moves (right). The HP sequence (HHPPHHPPHH) of monomers remain invariant and the chain remains intact which means the moves respect the conditions of the HP model and LCSAW.

testing since the configuration coordinates were printed after every pull and kink move with clear labelling as to what move caused the resulting configuration.

A series of chain pathways, as was done for the pull and pivot move algorithms, were generated. The chain starts linear and then a pull move is executed, then a kink flip move. This is done throughout the testing: first pull then perform a kink flip move.

The chain pathways presented in figures 23 and 24 are snippets of the sequence of 29 moves where the kink flip move was not rejected. The 6-mer (HHPPHH) was used aswell as with  $L = 300$  and random seed # 9062.

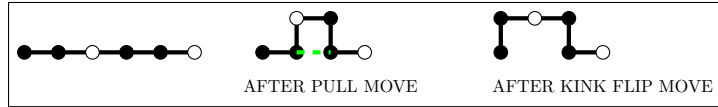


Figure 23: Simple chain pathway from linear chain  $\rightarrow$  pull moved chain  $\rightarrow$  kink of chain being flipped successfully.

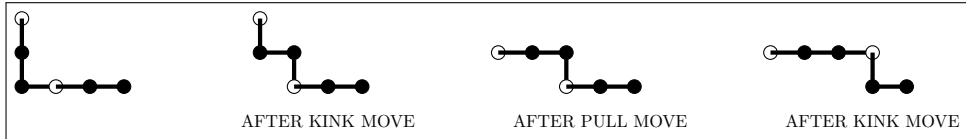


Figure 24: A longer sequence of successful pull and kink flip moves.

A larger test, as with the previous move tests, was conducted using a 10-mer (HHPPHH-PPHH). The sequence of moves was pull  $\rightarrow$  kink flip  $\rightarrow$  pull  $\rightarrow$  kink flip etc. for a total of 1000 moves. 500 kink flips and 500 pulls were conducted. In this test the same random seed # 9062 and lattice side length  $L = 300$  was used as before. The starting configuration and ending configuration shown in figure 25 was recorded.

Since the FRW and bond re-bridging moves were included in latter versions of the simulation code, extensive move testing was not conducted. However small trials were run to manually check (drawing out the configurations for small chains) the robustness and execution of the move algorithms.



Figure 25: The starting linear chain (left) and the resulting chain (right) after 1000 moves.

## Test Conclusions

From these basic tests it is clear that the trial set moves perform their intended operations on the chain. There is a possibility that the trial move sets can produce illegal chain configurations, since no human can predict how or when this will happen it is best to place a configuration barrier as outlined in subsection 3.5.2.

## 3.6 Energy Computing Routine

To register configurations which are in potential native states and to compute the total energy of the system using equation 1 for the Monte Carlo procedures, it is necessary to have an energy computing subroutine within the program. The routine needs to sum all the *topological* H-H contacts and the total energy of the system, using  $\epsilon_{HH} = 1$ , would simply be the negative of this sum. A monomer  $B$  is said to be the topological neighbour of monomer  $A$  when the 1D array coordinate of  $B$ ,  $B_\epsilon$ , is such that  $B_\epsilon \in \{A_\epsilon + 1, A_\epsilon - 1, A_\epsilon + L, A_\epsilon - L\}$  and when the sequence value of  $B$ ,  $s_B$ ,  $\neq s_A + 1$  and  $\neq s_A - 1$ .

The test of this routine, which was conducted early on in the development of the simulation, is shown in appendix B where native states of very short chained proteins are found, using a simple scoring system.

## 4 Energy Interval Experiment for WLS

Since in the Wang Landau sampling regime the reduction of the modification factor,  $f$ , directly dictates approximate convergence to the correct density of states, it is important to consider the energy ranges for the histogram. This consideration is unique to systems in which the difficulty of sampling configuration space grows with decreasing temperature.

In this protein folding model the difficulty in sampling dense low temperature configurations is known [11] [9] [35] [12] and when exploring the thermodynamic behaviour of folding and unfolding processes one has to strike a balance between convergence and exploring very deep wells in the energy landscape. This balance is a conflict between computational time and desire for detail.

Five simulations were run for the sequence 2D64 with different seeds and energy ranges see table 5.

| Run Number | Energy Range | $\ln[f_{final}]$             | Seed   |
|------------|--------------|------------------------------|--------|
| 1          | 0:(-38)      | $\approx 0.0002$             | 591418 |
| 2          | 0:(-30)      | $\approx 2 \cdot 10^{-180}$  | 655512 |
| 3          | 0:(-40)      | 0.5                          | 40824  |
| 4          | 0:(-25)      | $\approx 9 \cdot 10^{-1324}$ | 197881 |
| 5          | 0:(-37)      | $\approx 0.001$              | 251351 |

Table 5: For the energy ranges 0 is the upper bound also note that  $\ln[f_{initial}] = 1$  as outlined in section 3.1.

The specific heat results of run 1, 2, 4 and 5 are shown in figure 26. Observables from run 3 were omitted due to their drastic nature as the error bars were orders of magnitude larger than the results.

The entropy for the same runs is shown in figure 27.

### 4.1 Discussions and Remarks

The final modification factor is a sign of how well the WL sampling converged and as expected run 4 resulted in the lowest factor. Run 3 only had its modification factor reduced only once which reflects the difficulty WL sampling faces when encompassing the low temperature regions. Run 1 had a sub-par resulting modification factor and run 2 converged extremely well. It is not established whether the modification factor of run 4 took on  $1/t$  functionality. Run 5 has a final modification factor which is greater than run 1 whilst having a lower energy range. This fact could hint towards the inevitability of having more than a Gaussian threshold amount of runs for results to be statistically meaningful.

One can see that the worst converged simulation (run 5) in figure 26 underestimates the specific heat capacity for the sequence, even though the energy range is larger compared to run 2 and 4. The other curves are strikingly similar despite having varying sampling energy

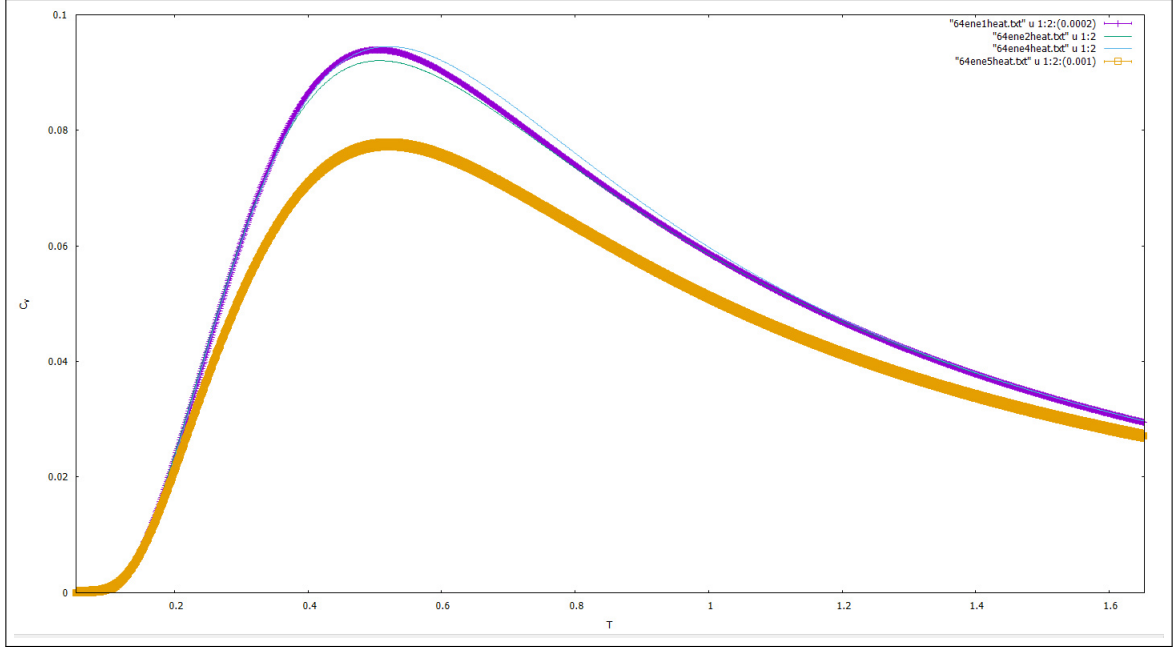


Figure 26: Purple = run 1, green = run 2, light blue = run 4 and gold = run 5. The error in  $C_v$  is the final modification factor shown in table 5, the errors for run 2 and 4 were omitted since they are smaller than the data points.

ranges. This could imply that cutting off the difficult, near native region, in the sampling is not as detrimental to the observables as was assumed.

Not surprisingly, for the entropy (see figure 27), one can see the difference between the observable computed from run 5 to the others, also note that all runs produced  $S < 0$  at very low temperatures (before  $T_c \approx 0.51$ ) which of course is not physically viable. This occurrence could be itself due to the lack of sampling of low temperature configurations mixed in with poor WL convergence.

This experiment has highlighted the need to take care in deciding the ultimate energy range for the WLS scheme for protein sequences. One needs to allow low temperature behaviour to be explored but without too much cost in accuracy. Also to attain decent modification factor reduction the routine must be run for a significant amount of time.

The lessons acquired from this small experiment were used in obtaining the final results.

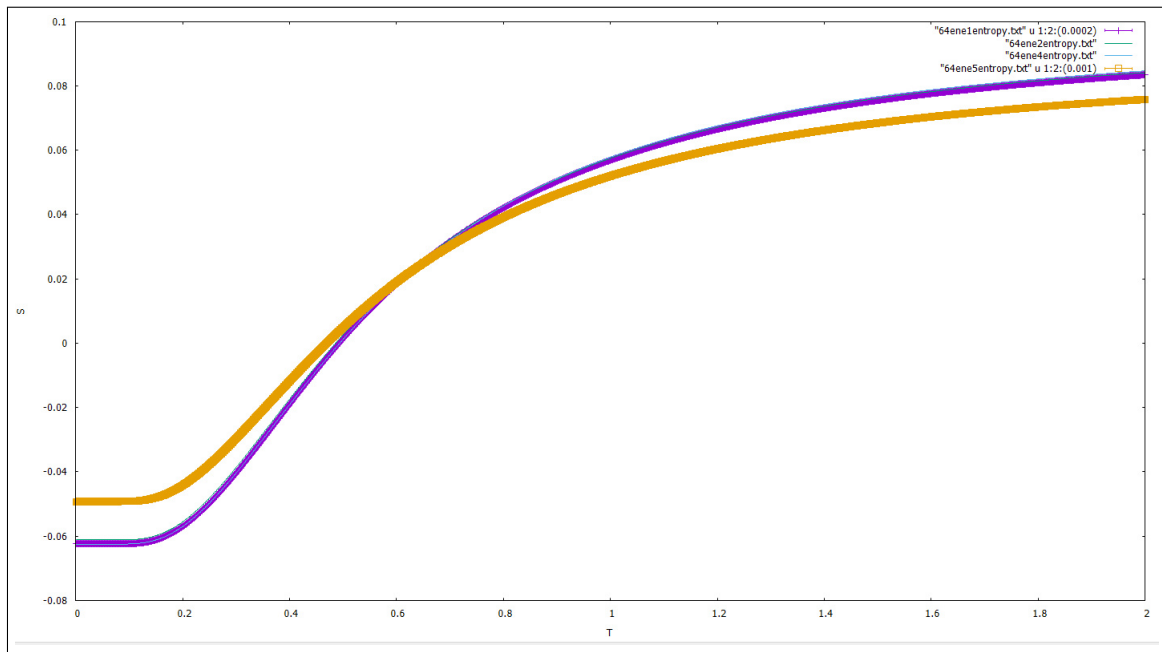


Figure 27: Purple = run 1, green = run 2, light blue = run 4 and gold = run 5. The error in  $S$  is the final modification factor shown in table 5, the errors for run 2 and 4 were omitted since they are smaller than the data points.



## 5 Results

## 5.1 Native State Search

With the following move set ratios: 65% pull, 19% bond re-bridging (of which 70% is type II), 10% fragment random walk, 4% pivot and 2% kink-flip, simulation runs were implemented with the specific aim of finding the native state of some benchmark sequences.

The sequences used in these runs were 2D50, 2D64 and 2D85. The (H)(P) sequence of these proteins are as follows:

**2D50 (S1-6)** HHPHPHPHPHHHPHPPPHPPPHPPPHPPPHPPPHPHPHHHPHPHPHPHH

2D60 (S1-7) PPHHHPPHHHHHHHHHHPPPHHHHHHHHHHHHPPPHHHHHHH  
HHHHHHPPPPHHHHHHHPHHHP

**2D64 (S1-8)** NNNNNNNNNNNNNRPHRPHRRNHRPNNHRRPHRPNHRRNHRPNNHRRPHRPNHRRNHR  
RPHRPHNNNNNNNNNNNN

**2D85 (S1-9)** HHHHPPRRHHHHHHHHHHHHHHHHHHHHPPRRPPRRHHHHHHHHHHHHHHHHHHHH  
HHHHPPRRHHHHHHHHHHHHHHHHHHHHPPRRHPPRHPPRHPPRHPPRH

2D100a (S1-10) PRRRRRHRNHRPPRRRRHNHRNNNNHRNHRPPRRRHNHRNHRNNNNHRNNNNH  
NNNNHRNHRNHHNNNNHPPRRRRPPRRRRNNNNNNHPPHRNHHRRPPRRRRHRNH

**2D100b (S1-11)** PRRPPRRHRNHRPPRRPNNHNRHHNNHNRRNHRPPRRNHRPNRHHNHN  
HNRRHHNNHHNNHHNNHRNHRNHHNNHHNNRRPPRRPPRRPNNHHNNHHNNRRHRNHHNRRPPRRPNNH

Results for the best minimum energy ( $E_{min}$ ) found compared to the best known native states from other methods are shown in table 6.

| Sequence | $E_{min}$ | WLS [9] | EMC [39] | SISPER [40] | GSA[41] | nPERMis [42] | EES [43] | FRESS [36] | ACO [46] |
|----------|-----------|---------|----------|-------------|---------|--------------|----------|------------|----------|
| 2D50     | -21       | N/A     | -21      | -21         | N/A     | N/A          | -21      | -21        | -21      |
| 2D60     | -36       | N/A     | -35      | -36         | -36     | -36          | -36      | -36        | -36      |
| 2D64     | -42       | -42     | -39      | -39         | -42     | -42          | -42      | -42        | -42      |
| 2D85     | -52       | -53     | N/A      | -52         | -52     | -53          | -53      | -53        | -53      |
| 2D100a   | -47       | -48     | N/A      | -48         | -48     | -48          | -48      | -48        | -47      |
| 2D100b   | -49       | -50     | N/A      | -49         | -50     | -50          | -49      | -50        | -49      |

Table 6: Comparison of native states found in this work (blue) with different methods .

The configuration for the native state of 2D50 and 2D64, found in this work, are shown in figures 28a and 28b respectively.

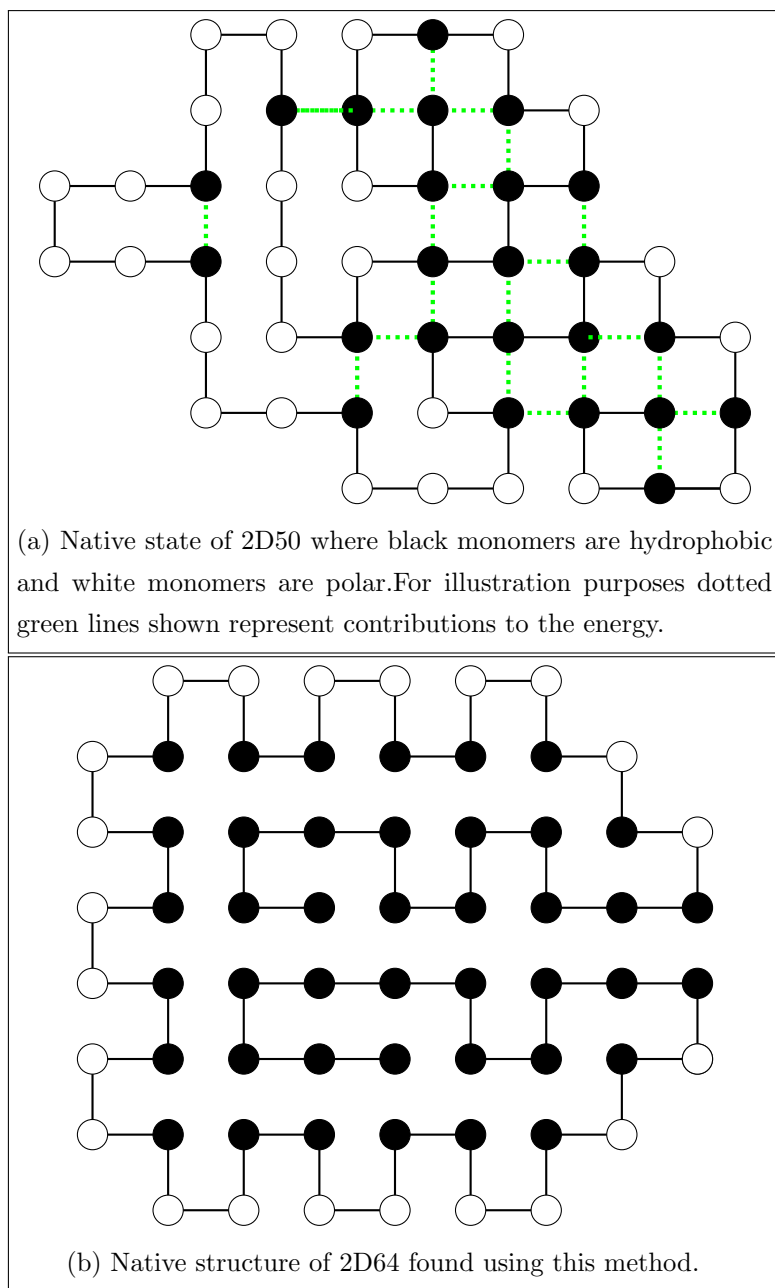


Figure 28: Example native structures.

## 5.2 Wang Landau Sampling

### 5.2.1 2D50

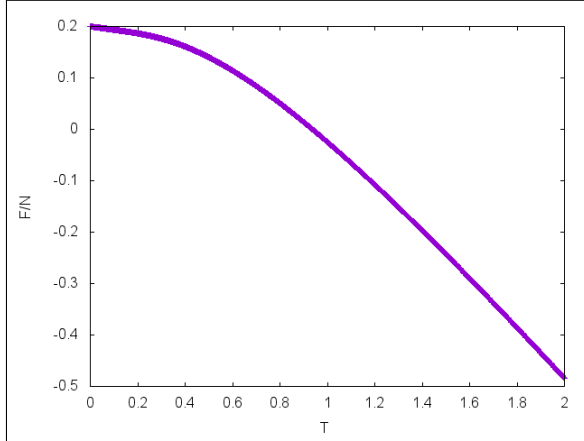
For the sequence 2D50 thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%, 19%, 5%, 4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature was found to be  $T_C = 0.576001$ . The final modification factor for each process is shown in table 7. Apart from process 2, 12 and 10 all reached the native state ( $E_{min} = -21$ ) and sampled it well. The energy range for the WLS was set to  $[-20 : 0]$ .

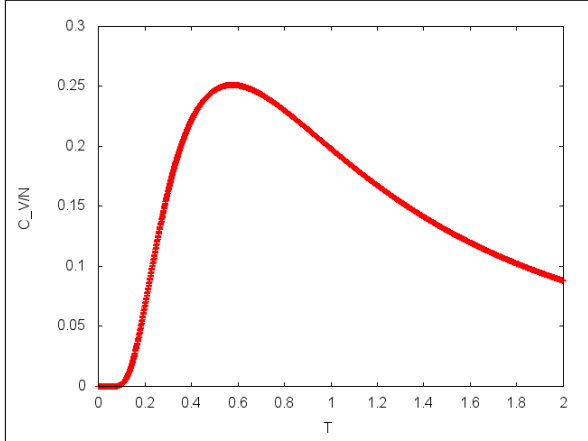
| Process ID | $\ln(f_{final})$             |
|------------|------------------------------|
| 0          | $\approx 2.38 \cdot 10^{-7}$ |
| 1          | $\approx 2.98 \cdot 10^{-8}$ |
| 2          | $\approx 2.98 \cdot 10^{-8}$ |
| 3          | $\approx 4.76 \cdot 10^{-7}$ |
| 4          | $\approx 2.38 \cdot 10^{-7}$ |
| 5          | $\approx 2.98 \cdot 10^{-8}$ |
| 6          | $\approx 1.49 \cdot 10^{-8}$ |
| 7          | $\approx 2.98 \cdot 10^{-8}$ |
| 8          | $\approx 1.19 \cdot 10^{-7}$ |
| 9          | $\approx 2.98 \cdot 10^{-8}$ |
| 10         | $\approx 1.49 \cdot 10^{-8}$ |
| 11         | $\approx 1.86 \cdot 10^{-9}$ |
| 12         | $\approx 7.45 \cdot 10^{-9}$ |
| 13         | $\approx 1.19 \cdot 10^{-7}$ |
| 14         | $\approx 1.19 \cdot 10^{-7}$ |

Table 7: The right column reflects the convergence of the intrinsic DOS for each process, the majority are  $\leq 10^{-7}$ , this convergence is adequate for the results shown in figure 29

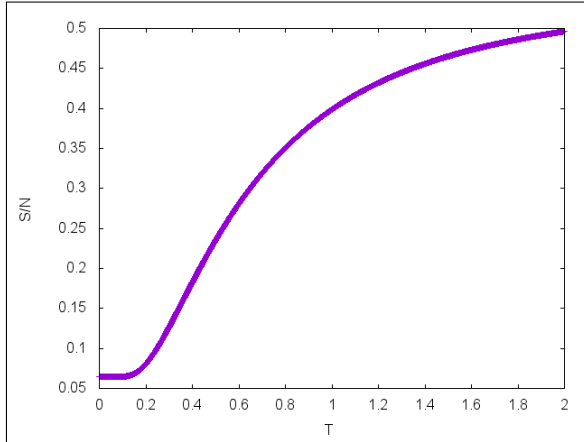
The Monte Carlo simulation for the following results completed  $\approx 2.7 \times 10^9$  iterations.



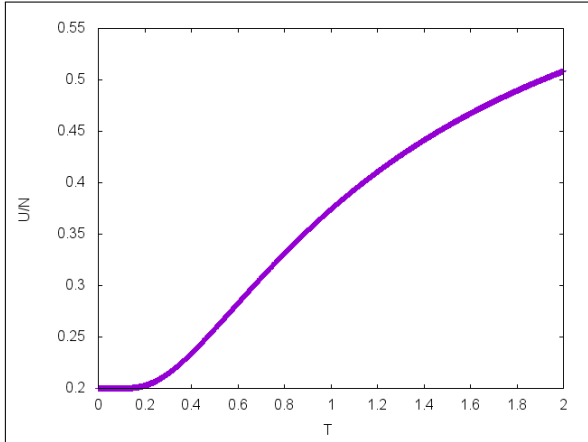
(a) The free energy  $F/N$  for 2D50. Error bars computed as described in 5.3.1.



(b) The specific heat capacity  $C_V/N$  for 2D50. Error bars computed as described in 5.3.1.



(c) The entropy  $S/N$  for 2D50. Error bars computed as described in 5.3.1.



(d) The internal energy  $U/N$  for 2D50. Error bars computed as described in 5.3.1.

Figure 29: Computed thermodynamic observables for 2D50.

### 5.2.2 2D60

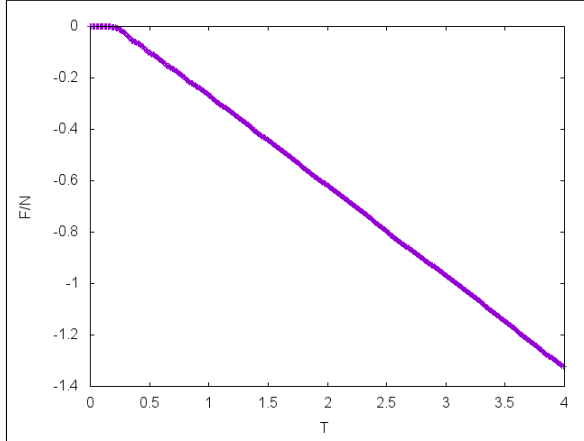
For the sequence 2D60 thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%,19%, 5%,4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature was found to be  $T_C = 0.42$ . The final modification factor for each process is shown in table 8. The achievement of accessing the native state (-36) of 2D60 was accomplished during this run. Only process 6 achieved this state and the others reached a minimum of (-35). The energy range for the sampling was set to [-34:00] for these results.

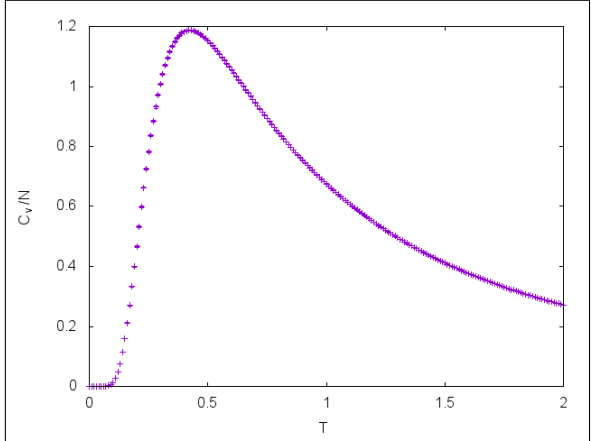
| Process ID | $\ln(f_{final})$               |
|------------|--------------------------------|
| 0          | $\approx 3.50 \cdot 10^{-46}$  |
| 1          | $\approx 8.55 \cdot 10^{-50}$  |
| 2          | $\approx 4.7 \cdot 10^{-38}$   |
| 3          | $\approx 4.27 \cdot 10^{-50}$  |
| 4          | $\approx 1.34 \cdot 10^{-51}$  |
| 5          | $\approx 7.45 \cdot 10^{-9}$   |
| 6          | $\approx 8.55 \cdot 10^{-50}$  |
| 7          | $\approx 3.85 \cdot 10^{-34}$  |
| 8          | $\approx 1.15 \cdot 10^{-41}$  |
| 9          | $\approx 3.58 \cdot 10^{-43}$  |
| 10         | $\approx 9.183 \cdot 10^{-41}$ |
| 11         | $\approx 5.72 \cdot 10^{-42}$  |
| 12         | $\approx 8.758 \cdot 10^{-47}$ |
| 13         | $\approx 1.54 \cdot 10^{-33}$  |
| 14         | $\approx 2.08 \cdot 10^{-53}$  |

Table 8: The right column reflects the convergence of the intrinsic DOS for each process, the majority are  $\leq 10^{-30}$ , this convergence is adequate for the results shown in figure 30.

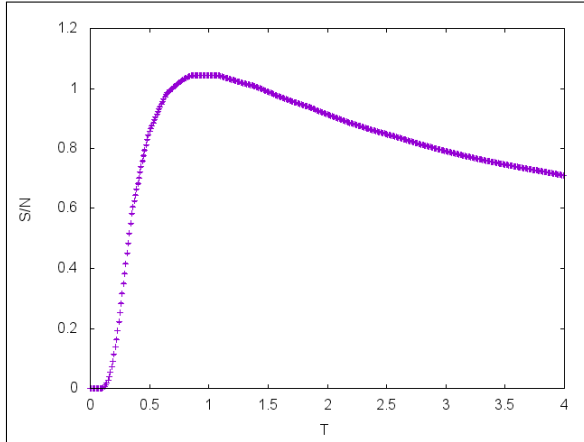
The Monte Carlo simulation for the following results completed  $\approx 5.8 \times 10^8$  iterations.



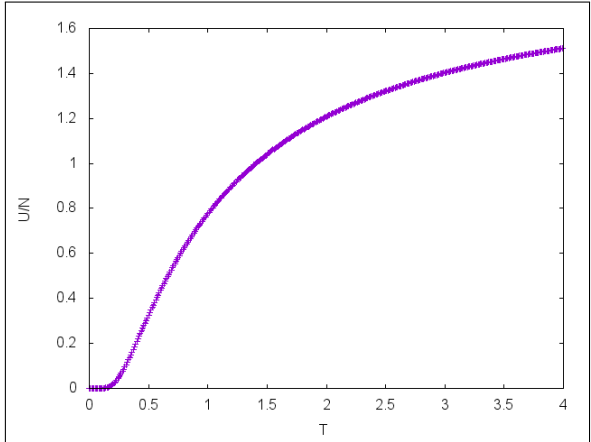
(a) The free energy  $F/N$  for 2D60. Error bars computed as described in 5.3.1.



(b) The specific heat capacity  $C_V/N$  for 2D60. Error bars computed as described in 5.3.1.



(c) The entropy  $S/N$  for 2D60. Error bars computed as described in 5.3.1.



(d) The internal energy  $U/N$  for 2D60. Error bars computed as described in 5.3.1.

Figure 30: Computed thermodynamic observables for 2D60.

### 5.2.3 2D64

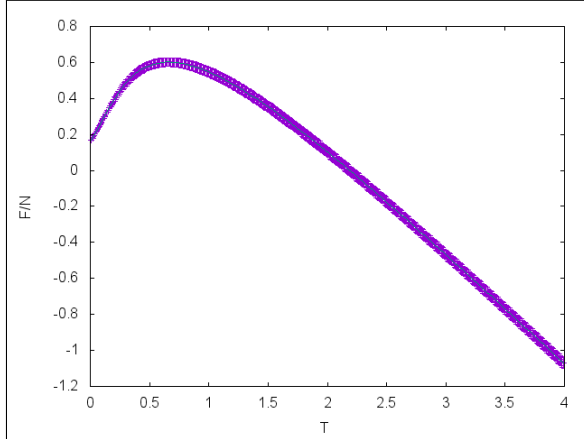
For the sequence 2D64 thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%,19%, 5%,4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature was found to be  $T_C = 0.39$ . The final modification factor for each process is shown in table 8. During this short simulation every process attained the minimum energy of -40 which was set to the lower bound of the WL energy range.

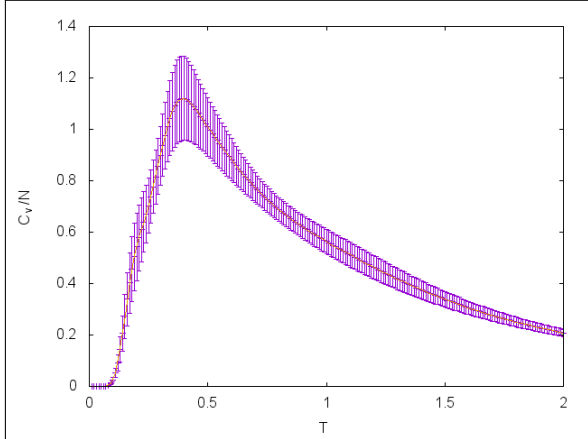
| Process ID | $\ln(f_{final})$              |
|------------|-------------------------------|
| 0          | $\approx 1.563 \cdot 10^{-2}$ |
| 1          | $\approx 0.313$               |
| 2          | $\approx 0.01563$             |
| 3          | $\approx 0.01563$             |
| 4          | $\approx 0.0078$              |
| 5          | $\approx 0.01563$             |
| 6          | $\approx 0.01563$             |
| 7          | $\approx 0.007813$            |
| 8          | $\approx 0.03125$             |
| 9          | $\approx 0.015625$            |
| 10         | $\approx 0.007813$            |
| 11         | $\approx 0.0313$              |
| 12         | $\approx 0.0313$              |
| 13         | $\approx 0.007813$            |
| 14         | $\approx 0.0313$              |

Table 9: The right column reflects the convergence of the intrinsic DOS for each process. This convergence is questionably adequate for the results shown in figure 31 (see section 6 for an explanation).

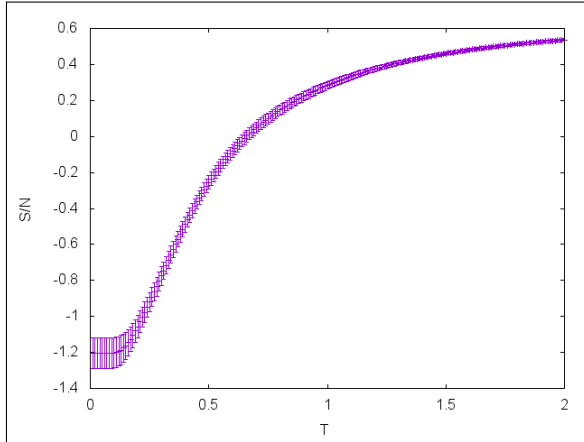
The Monte Carlo simulation for the following results completed  $\approx 808 \times 10^6$  iterations.



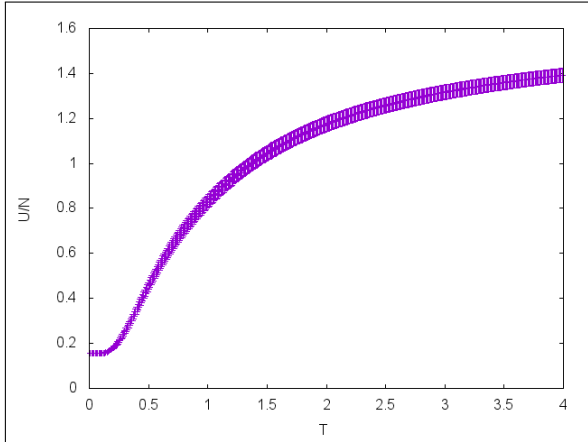
(a) The free energy  $F/N$  for 2D64. Error bars computed as described in 5.3.1.



(b) The specific heat capacity  $C_V/N$  for 2D64. Error bars computed as described in 5.3.1.



(c) The entropy  $S/N$  for 2D64. Error bars computed as described in 5.3.1.



(d) The internal energy  $U/N$  for 2D64. Error bars computed as described in 5.3.1.

Figure 31: Computed thermodynamic observables for 2D64.



### 5.2.4 2D85

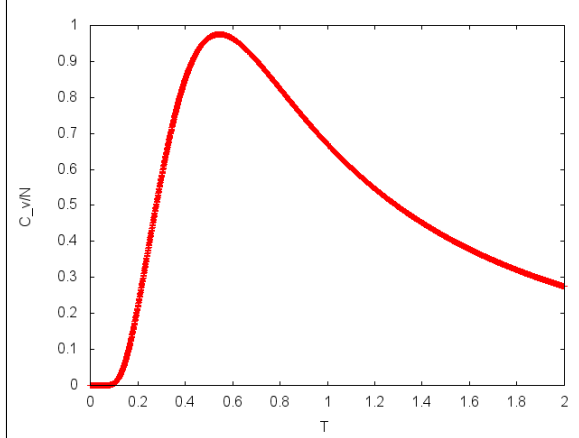
For the sequence 2D85 thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%,19%, 5%,4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature, at which  $C_V/N$  is a maximum, was found to be  $T_C = 0.545001$ . The final modification factor for each process is shown in table 10. All processes reached a minimum of -51 which was used as the lower limit of the energy range.

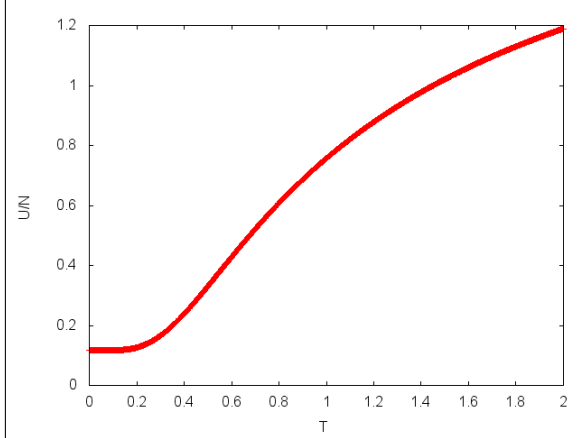
| Process ID | $\ln(f_{final})$             |
|------------|------------------------------|
| 0          | $\approx 1.22 \cdot 10^{-4}$ |
| 1          | $\approx 1.91 \cdot 10^{-6}$ |
| 2          | $\approx 7.63 \cdot 10^{-6}$ |
| 3          | $\approx 7.63 \cdot 10^{-6}$ |
| 4          | $\approx 1.91 \cdot 10^{-6}$ |
| 5          | $\approx 3.81 \cdot 10^{-6}$ |
| 6          | $\approx 1.91 \cdot 10^{-6}$ |
| 7          | $\approx 9.54 \cdot 10^{-7}$ |
| 8          | $\approx 1.91 \cdot 10^{-6}$ |
| 9          | $\approx 3.81 \cdot 10^{-6}$ |
| 10         | $\approx 1.91 \cdot 10^{-6}$ |
| 11         | $\approx 3.81 \cdot 10^{-6}$ |
| 12         | $\approx 4.7 \cdot 10^{-7}$  |
| 13         | $\approx 1.91 \cdot 10^{-6}$ |
| 14         | $\approx 1.22 \cdot 10^{-4}$ |

Table 10: The right column reflects the convergence of the intrinsic DOS for each process, the majority are  $< 10^{-5}$ , this convergence is adequate for the results shown in figures 32.

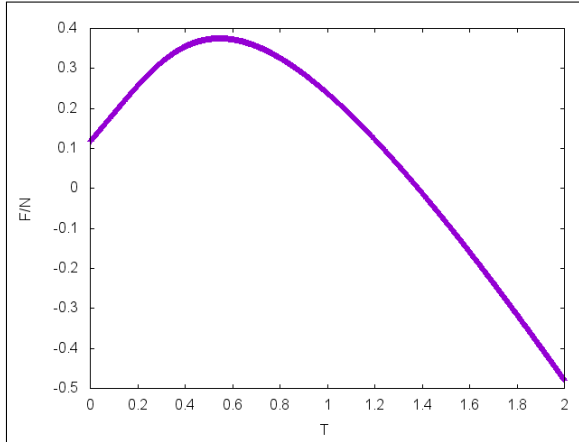
The Monte Carlo iterations for this simulation run was = 1347840000.



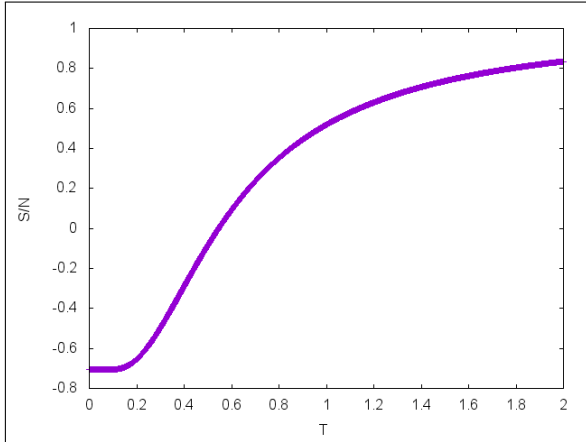
(a) Specific heat capacity,  $C_V$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(b) Internal energy,  $U$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(c) Free energy,  $F$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(d) Entropy,  $S$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.

Figure 32: Thermodynamic observables for 2D85.

### 5.2.5 2D100a

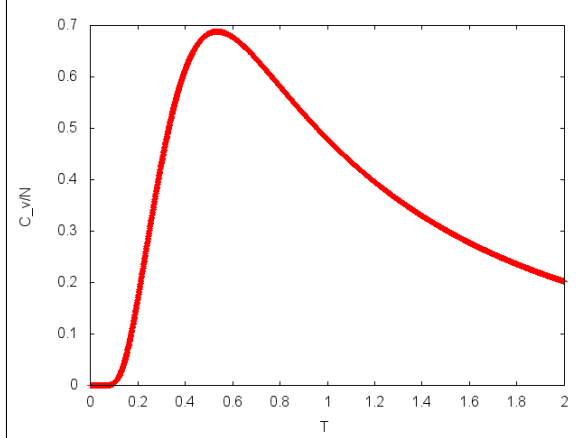
For the sequence 2D100a thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%, 19%, 5%, 4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature was found to be  $T_C = 0.535$ . The final modification factor for each process is shown in table 11. Every process attained a minimum energy = -47 which is a unit of energy greater than the lowest known (see 6). The energy range was then, automatically, set to [0:-47] (the global range being [0:-48] but the processes only attained -47).

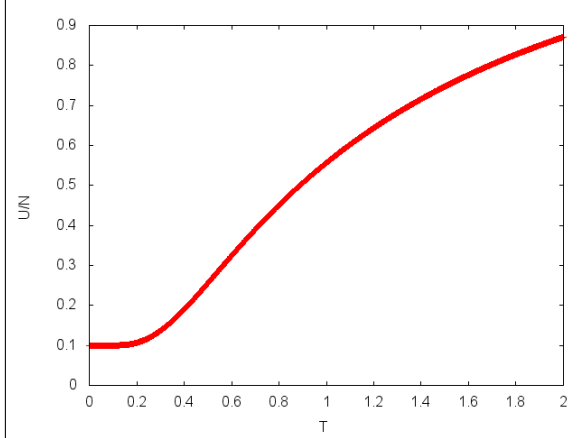
| Process ID | $\ln(f_{final})$              |
|------------|-------------------------------|
| 0          | $\approx 3.82 \cdot 10^{-6}$  |
| 1          | $\approx 3.82 \cdot 10^{-6}$  |
| 2          | $\approx 7.63 \cdot 10^{-6}$  |
| 3          | $\approx 3.05 \cdot 10^{-5}$  |
| 4          | $\approx 1.53 \cdot 10^{-5}$  |
| 5          | $\approx 7.63 \cdot 10^{-6}$  |
| 6          | $\approx 3.82 \cdot 10^{-6}$  |
| 7          | $\approx 3.82 \cdot 10^{-6}$  |
| 8          | $\approx 3.82 \cdot 10^{-6}$  |
| 9          | $\approx 3.052 \cdot 10^{-5}$ |
| 10         | $\approx 7.63 \cdot 10^{-6}$  |
| 11         | $\approx 1.53 \cdot 10^{-5}$  |
| 12         | $\approx 1.91 \cdot 10^{-6}$  |
| 13         | $\approx 3.052 \cdot 10^{-5}$ |
| 14         | $\approx 3.815 \cdot 10^{-6}$ |

Table 11: The right column reflects the convergence of the intrinsic DOS for each process, the majority are  $< 10^{-5}$ , this convergence is adequate for the results shown in figure 33

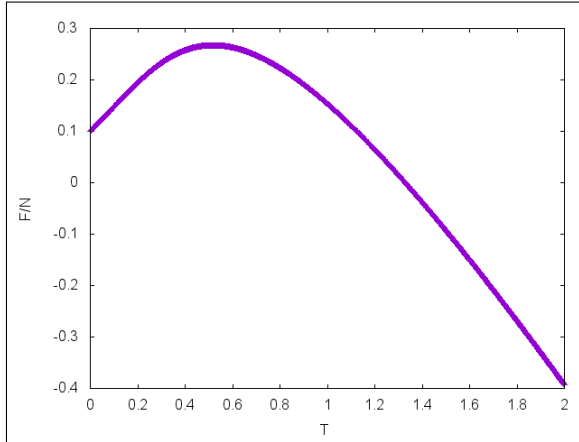
The Monte Carlo iterations for this simulation run was = 1347840000.



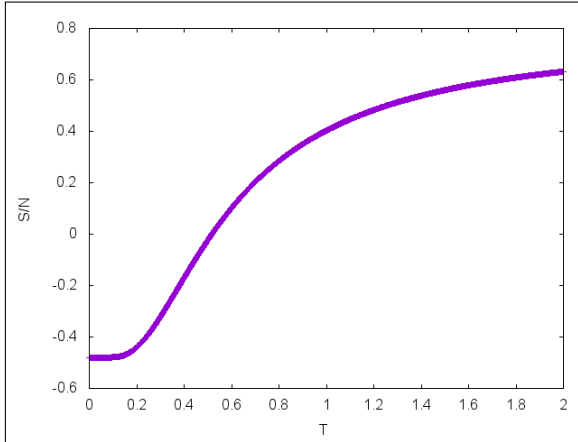
(a) Specific heat capacity,  $C_V$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(b) Internal energy,  $U$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(c) Free energy,  $F$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(d) Entropy,  $S$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.

Figure 33: Thermodynamic observables for 2D100a.

### 5.2.6 2D100b

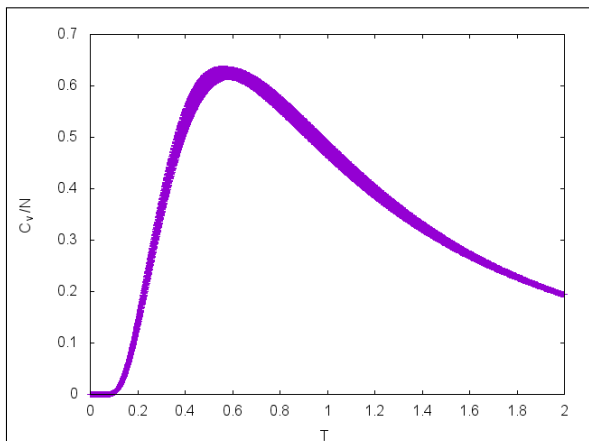
For the sequence 2D100b thermodynamic behaviour was investigated via the computation of  $C_V/N$ ,  $U/N$ ,  $S/N$  and  $F/N$ . The flatness criterion for this simulation was  $p = 0.8$  and the move ratios were 70%, 19%, 5%, 4% and 2% for pull, bond re-bridging, FRW, pivot and kink-flip moves respectively.

The 'critical' temperature was found to be  $T_C = 0.5765 \pm 0.02$ . The final modification factor for each process is shown in table 12. Each process attained the energy of -46 which is 4 more than the known native state of 100b.

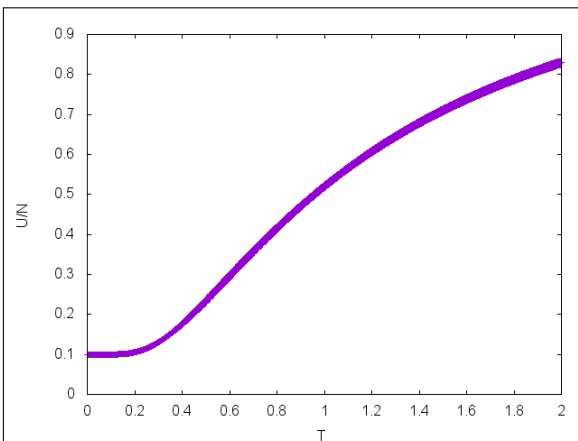
| Process ID | $\ln(f_{final})$             |
|------------|------------------------------|
| 0          | $\approx 3.91 \cdot 10^{-3}$ |
| 1          | $\approx 0.0039$             |
| 2          | $\approx 0.0078$             |
| 3          | $\approx 0.00097$            |
| 4          | $\approx 0.0039$             |
| 5          | $\approx 0.0039$             |
| 6          | $\approx 0.00195$            |
| 7          | $\approx 0.0039$             |
| 8          | $\approx 0.0039$             |
| 9          | $\approx 0.0078$             |
| 10         | $\approx 0.00098$            |
| 11         | $\approx 0.00195$            |
| 12         | $\approx 0.00195$            |
| 13         | $\approx 0.0039$             |
| 14         | $\approx 0.0039$             |

Table 12: The right column reflects the convergence of the intrinsic DOS for each process, the majority are  $< 0.01$ . Observables for this run are shown in figure 34.

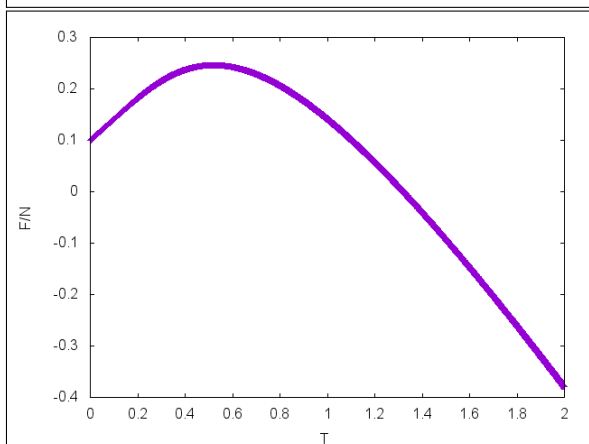
The Monte Carlo iterations for this simulation run was = 1347840000.



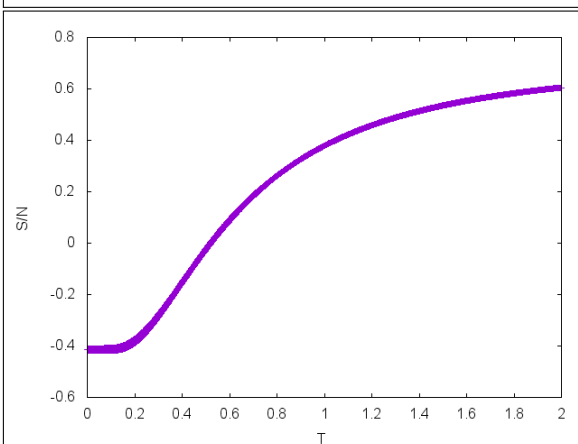
(a) Specific heat capacity,  $C_V$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(b) Internal energy,  $U$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(c) Free energy,  $F$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.



(d) Entropy,  $S$ , divided by the number of monomers,  $N$ , against  $T$ . Error bars computed as described in 5.3.1.

Figure 34: Thermodynamic observables for 2D100b.

### 5.3 ISAWs

Homopolymers, which consist of identical sub units, are found in industrial plastics and biology. Using the methodology (see section 3) which was used to investigate the behaviour of HP proteins, ISAWs (Interacting Self Avoiding Walks) of Homopolymers were also studied. Each sub unit of the homopolymer is assumed to be hydrophobic and the energy function given by equation 1.

Homopolymers of lengths: 25, 64, 100, 144, 225 and 400 were studied and the convergence of the modification factors for each sequence and the comparison of thermodynamic behaviour is given in table 13 and figures 35 and 36 respectively. The ratios of the trial moves were exactly the same as in the previous section. The flatness criterion was  $p = 0.8$  for lengths 25,64, 100, 144 and  $p = 0.7$  for lengths 225 and 400 due to difficulty in traversing the conformational space in the given time length.

| Process ID | $\ln(f_{final})$                 |
|------------|----------------------------------|
| 0          | $\approx 3.953 \cdot 10^{-4327}$ |
| 1          | $\approx 3.287 \cdot 10^{-4438}$ |
| 2          | $\approx 1.412 \cdot 10^{-3785}$ |
| 3          | $\approx 4.707 \cdot 10^{-4188}$ |
| 4          | $\approx 3.298 \cdot 10^{-4233}$ |
| 5          | $\approx 9.246 \cdot 10^{-2992}$ |
| 6          | $\approx 6.324 \cdot 10^{-4326}$ |
| 7          | $\approx 2.871 \cdot 10^{-3403}$ |
| 8          | $\approx 1.867 \cdot 10^{-4305}$ |
| 9          | $\approx 2.815 \cdot 10^{-3990}$ |

| Process ID | $\ln(f_{final})$                |
|------------|---------------------------------|
| 0          | $\approx 2.328 \cdot 10^{-10}$  |
| 1          | $\approx 2.910 \cdot 10^{-11}$  |
| 2          | $\approx 2.91 \cdot 10^{-11}$   |
| 3          | $\approx 3.725 \cdot 10^{-9}$   |
| 4          | $\approx 1.863 \cdot 10^{-9}$   |
| 5          | $\approx 7.451 \cdot 10^{-9}$   |
| 6          | $\approx 4.768 \cdot 10^{-7}$   |
| 7          | $\approx 1.863 \cdot 10^{-9}$   |
| 8          | $\approx 4.6567 \cdot 10^{-10}$ |
| 9          | $\approx 4.6567 \cdot 10^{-10}$ |

(a) Convergence for each process for length **25**. (b) Convergence for each process for length **64**.

| Process ID | $\ln(f_{final})$               |
|------------|--------------------------------|
| 0          | $\approx 6.104 \cdot 10^{-5}$  |
| 1          | $\approx 9.313 \cdot 10^{-10}$ |
| 2          | $\approx 1.192 \cdot 10^{-7}$  |
| 3          | $\approx 2.980 \cdot 10^{-8}$  |
| 4          | $\approx 1.526 \cdot 10^{-5}$  |
| 5          | $\approx 9.313 \cdot 10^{-10}$ |
| 6          | $\approx 9.313 \cdot 10^{-10}$ |
| 7          | $\approx 2.980 \cdot 10^{-8}$  |
| 8          | $\approx 9.131 \cdot 10^{-10}$ |
| 9          | $\approx 3.725 \cdot 10^{-9}$  |

| Process ID | $\ln(f_{final})$              |
|------------|-------------------------------|
| 0          | $\approx 3.125 \cdot 10^{-2}$ |
| 1          | $\approx 0.03125$             |
| 2          | $\approx 0.03125$             |
| 3          | $\approx 0.007813$            |
| 4          | $\approx 0.015625$            |
| 5          | $\approx 0.015625$            |
| 6          | $\approx 0.015625$            |
| 7          | $\approx 0.0625$              |
| 8          | $\approx 0.03125$             |
| 9          | $\approx 0.03215$             |

(c) Convergence for each process for length **100**. (d) Convergence for each process for length **144**.

| Process ID | $\ln(f_{final})$              |
|------------|-------------------------------|
| 0          | $\approx 7.813 \cdot 10^{-3}$ |
| 1          | $\approx 0.007813$            |
| 2          | $\approx 0.007813$            |
| 3          | $\approx 0.007813$            |
| 4          | $\approx 0.01563$             |
| 5          | $\approx 0.01563$             |
| 6          | $\approx 0.01563$             |
| 7          | $\approx 0.01563$             |
| 8          | $\approx 0.01563$             |
| 9          | $\approx 0.03125$             |

| Process ID | $\ln(f_{final})$ |
|------------|------------------|
| 0          | $\approx 1$      |
| 1          | $\approx 1$      |
| 2          | $\approx 1$      |
| 3          | $\approx 1$      |
| 4          | $\approx 1$      |
| 5          | $\approx 1$      |
| 6          | $\approx 1$      |
| 7          | $\approx 1$      |
| 8          | $\approx 1$      |
| 9          | $\approx 1$      |

(e) Convergence for each process for length **225**. (f) 'Convergence' for each process for length **400**.

Table 13: Final modification factors for ISAW length simulations.

| ISAW length | Total MC iterations | Duration (s) |
|-------------|---------------------|--------------|
| 25          | $2695 \times 10^6$  | 80386        |
| 64          | $439 \times 10^6$   | 89450        |
| 100         | $236 \times 10^6$   | 92119        |
| 144         | $124 \times 10^6$   | 100427       |
| 225         | $48 \times 10^6$    | 90609        |
| 400         | $31 \times 10^6$    | 171526       |

Table 14: Monte Carlo iterations and duration of simulation runs.

A comparison of  $C_v/N$  and  $U/N$  for all lengths except 144, 225 and 400, since the simulations did not converge adequately (error bars greater than the results), as a function of temperature is shown in figures 35 and 36 respectively. The errors are computed following the descriptions in section 5.3.1.

The minimum energies attained, which represent the lower boundary of the WL energy range for the ISAWs, are compared to that found by [51] in figure 37.



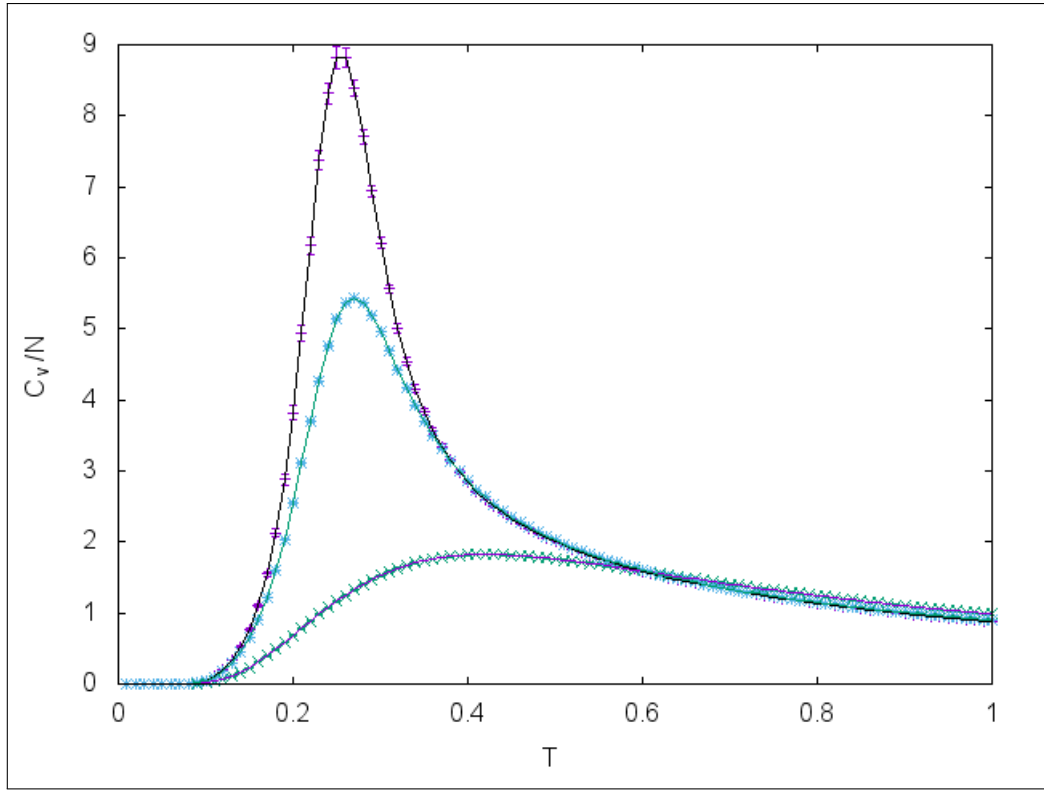


Figure 35: Specific heat capacity per monomer,  $C_v/N$ , against temperature  $T$ . Length 25 (purple, green error bars) = lower curve, length 64 (green, blue errorbars) = middle curve and length 100 (black, purple error bars) = highest curve.

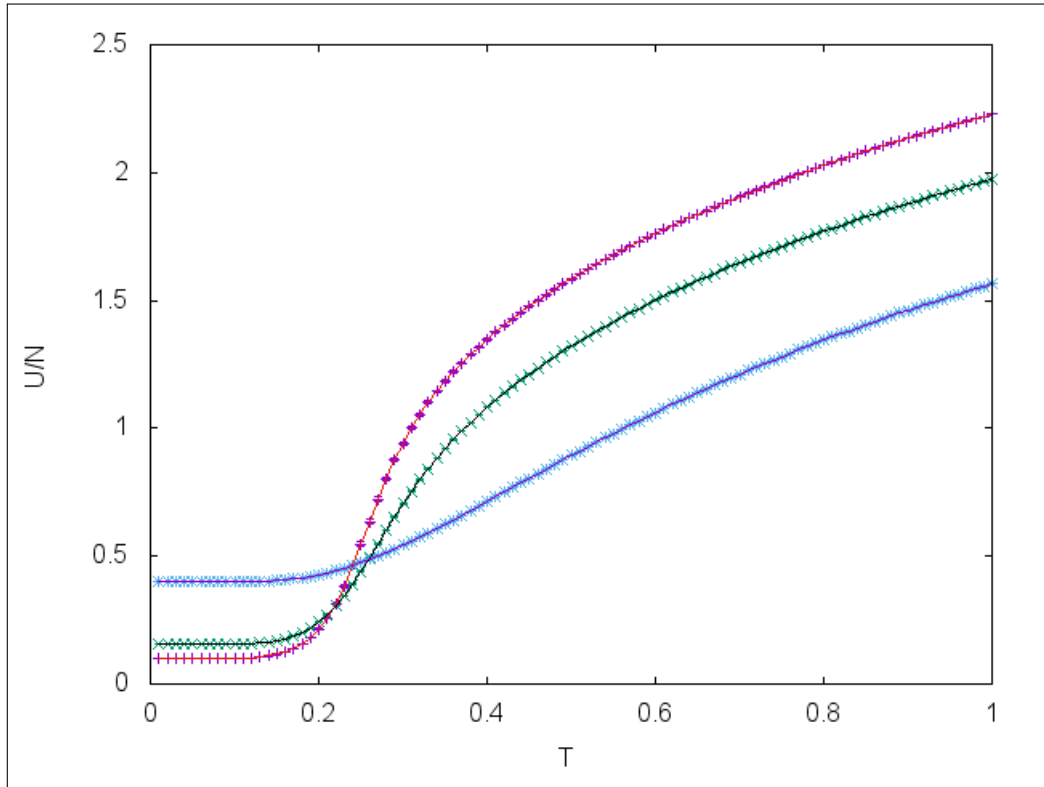


Figure 36: Internal energy per monomer,  $U/N$ , against temperature  $T$ . Length 25 (blue error bars), length 64 (green errorbars) and length 100 (purple error bars) = highest curve.

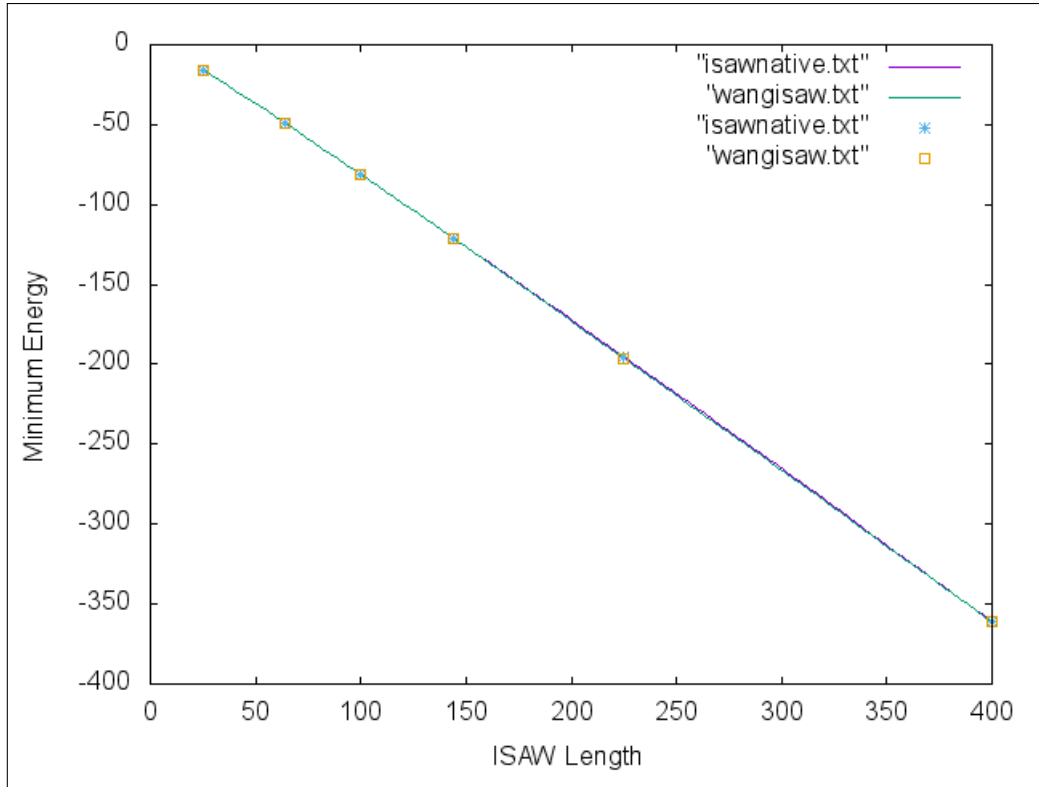


Figure 37: Graphical comparison of minimum energies found for benchmark ISAWs. N.B. simulation timings for the Wust-Landau results unknown. 'isawnative' are results from this work and 'wangisaw' are results taken from [51].

### 5.3.1 Error Analysis

It has been stated that the general uncertainty in the computed DOS is  $\propto f$  (the modification factor) 3.1. However since thermodynamic observables were computed using the parallel-trajectory- swapping scheme where many random walkers are used computing their own observables and averages were taken, it is necessary to consider statistical variations centred about the mean<sup>13</sup>.

The variance,  $s^2$ , of  $n$  observations  $\{x_1, x_2, \dots, x_n\}$  is:

$$s^2 = \frac{1}{n-1} [(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \dots + (x_n - \bar{x})^2] \quad (28)$$

where  $\bar{x}$  is the sample mean of the data set [38]. The standard deviation,  $\sigma$ , is simply given by the square root of  $s^2$ .

When the standard deviation of a statistic is estimated from the data this is the standard error,  $SE$ , [38], which is the error used here:

$$SE = \frac{\sigma}{\sqrt{n}} \quad (29)$$

For each temperature the thermodynamic observables e.g.  $C_V$  from each WL walker were averaged and the resulting error was computed using equation 29 [8].

---

<sup>13</sup>Results obtained from single walkers only have error bars centred around the modification factor.

## 6 Discussion of Results

### 6.0.2 Native State Search

Attaining the native state of 2D64 is known to be difficult, for example the computational methods of EMC and SISPER only could attain  $E_{min} = -39$  [36] (see table 6). However this simulation method has not only reached the lowest known energy of this sequence but also found a unique hydrophobic core for the native structure of 2D64 (see figure 28b). A visual comparison of the native structures found here and with WLS [9] and ACO [46] is shown in figure 38. One can then see that the particular external structure of 2D64 is exact, which explains why it is difficult to access the native region since there are few native structures.

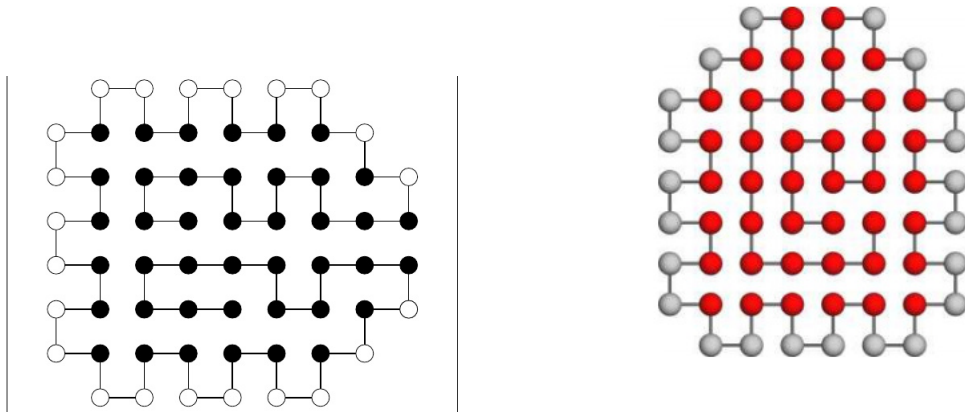
This striking similarity between found native structures for 2D64 reinforces the notion that proteins fold into specific structures in order to execute the same physiological function.

The native structures for 2D50 and 2D64 found in this particular run (see figure 28a) has confirmed the expectation that most of the hydrophobic amino acids push towards the center of the compact configuration, leaving the polar amino acids to bond with the external aqueous solution. Also compare the native state found in the previous section to appendix B (2D50) where the best  $E_{min}$  was -13. This vast improvement arose due to the inclusion of the bond re-bridging and FRW move and an improved move ratios.

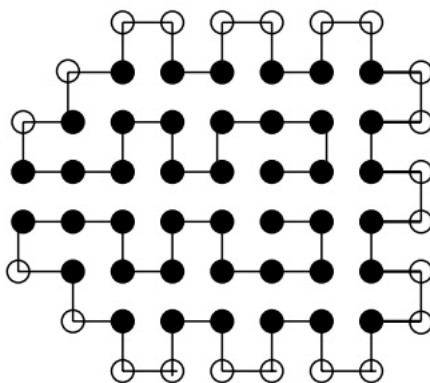
Note that the native structure for 2D50 found here (see figure 28a) contains a few polar amino acids 'locked' in the hydrophobic center. Recall that ionic bonds are formed as amino acids bearing opposite electrical charge are close together within the hydrophobic core of proteins. Ionic bonding in the hydrophobic core is rare since most charged amino acids are polar, which are normally pushed towards the edge of the protein surface due to the hydrophobic effect. Although rare, ionic bonds can play an important role in stabilizing the native structure that can approach the strength of covalent bonding.

The inability to access the lowest known energies of 2D85 (-53), 2D100a (-48) and 2D100b (-50) does not reflect an intrinsic limitation of this method or trial set. Since WLS [9] attained these native structures with a similar but ultimately distinct trial move set, it is very likely that this method can also in principle access these low temperature structures. I believe it is a matter of insufficient computational effort and time that limited accessing these energies. The simulations here attained energies 1 unit more than the native state for the sequences just mentioned, which in terms of exploring global thermodynamic behaviour is not a complete downfall. The reader is encouraged to see section 4 where it was demonstrated that thermodynamic observables are not greatly changed if low energies become unavailable to the WL sampler.

Further optimisation and computational 'tinkering' could improve the efficiency of the code used here which would be able to directly compete with WLS [9].



(a) 2D64 native structure found in this work. (b) 2D64 native structure found in WLS (Wust-Landau)[9].



(c) 2D64 native structure found in the ACO method [46].

Figure 38: Comparison of native structures for 2D64. The similarity of the external polar amino acid placement is striking, also each sequence has the same line of symmetry (externally). The difference between the native structures is found within the hydrophobic core, however this will not necessarily alter the function of the protein since it interacts with others via its external structure.

## ISAWs

As shown in figure 37 the lowest energy states for the same length of ISAWs in this work compared well with [51]. For  $N > 144$  the lowest energy states found in this work were slightly higher than [51], this is notably to the huge computational burden long chains present. Since all ISAW sequences were run for a similar amount of time in seconds (See table 14) the amount of MC iterations for longer chains obviously decreased. This would mean potentially less coverage of conformational space and hence not accessing the native configuration.

ISAWs present a distinct problem in accessing low temperature structures since they exist in sharp wells in the rough energy landscape, using Wang- Landau sampling, trajectory swapping and the trial move set proposed here allows quick and thorough coverage of conformational

space and with enough computational effort could access extremely long ISAWs and their native structures.

### 6.0.3 Thermodynamic Investigations

The general thermodynamic observables, the specific heat capacity in particular, show a 'pseudo phase transition'<sup>14</sup> at a particular critical temperature  $T_C$ . The following discussion of thermodynamic and protein behaviour will consist of first the high-T regime and then the low-T regime.

$$\underline{T > T_C}$$

The specific heat capacity,  $C_V/N$ , computed for all 2D benchmark protein (H)(P) sequences shows a gradual increase as the temperature goes from very high to just above the critical temperature  $T_C$  (e.g. see figure 29(b)). This happens as the chain goes from almost a 'string-like' configuration at high T and increasing the amount of H-H bonds as the temperature decreases. Going from a denatured state  $\rightarrow$  molten globule occurs during the temperatures just above the critical temperature.

All the 2D sequences, despite varying degrees of convergence, also show similar behaviour for  $U/N$ , which decreases at a gradually faster rate (closer to  $T_C$ ). This can easily be explained due to the increase in thermal agitation of the monomers on the chain as the temperature increases. Even for the accurate results for 2D50 and 2D60 there is no obvious sudden collapse of this thermodynamic observable.

The free energy,  $F/N$ , in this temperature regime grows almost linearly with decreasing temperature. This linear relationship is shown almost perfectly for 2D60 (See figure 30). Towards the critical region this growth in free energy gradually slows down for all sequences. The growth in free energy is again due to the increased number of H-H contacts and the protein becoming more globule-like, increasing the amount of thermodynamic work it can perform.

The entropy,  $S/N$ , behaves very much like the internal energy with temperature for all but 2D60. The entropy for 2D50, 2D64, 2D85, 2D100a and 2D100b decreases gradually (at a faster rate towards  $T_C$ ). The entropy for 2D50 (see figure 29) can be taken as an accurate representation for this entropy behaviour since it converged better than the other 4 sequences. The gradual decrease in entropy aligns with our thermodynamic expectations since the degeneracy for configurations at lower temperatures decreases. These results also meet the expectation that the system will be in a swollen SAW state where entropy should dominate [52].

2D60 (the best converged simulation) showed an entropy that contained a peak near  $T_C = 0.42$  (see figure 30). The entropy very slightly increases with decreasing temperature towards its peak, this behaviour contrasts the results for the other sequences where the maximum entropy occurs at the highest temperature for the simulation. A physical explanation for this behaviour could be that at extremely high temperatures the chain becomes close to or attains

---

<sup>14</sup>'Pseudo' since the system is finite in size.

a straight line conformation on the lattice, where the degeneracy for this drops slightly. The entropy also seems to be converging to a value much  $> 0$ , so it is not expected that the entropy will continue to decrease with temperature which would not be physically viable. Whether this behaviour is sequence, length or convergent (final modification value) dependent unfortunately cannot be ascertained from the results here. However it does raise the question: Does the entropy always increase as  $T$  increases for every sequence? Also the entropy for 2D60 does signal a clear psuedo phase transition which could mean that the entropy can play a role as a 'phase transition signaller' for lattice polymers.

In general the thermodynamic observables in the  $T > T_C$  region do confirm the expectation that at high temperatures the protein chain becomes denatured and tends toward a rigid straight line configuration. The smoothness of the increase or decrease of observables (increasing rate towards  $T_C$ ) with decreasing temperature reflects the gradual increase of H-H topological contacts which are leading the chain into a globule-like configuration.

$$\underline{T \leq T_C}$$

The specific heat capacity for all sequences change drastically as the temperature passes through  $T_C$  to lower values. The gradient of the observable at  $T < T_C$  is of opposite sign to that at  $T > T_C$ , also the absolute magnitude of the gradient is larger due to the fast rate of decline as  $T \rightarrow 0$ . This behaviour reflects the chain attaining highly compact configurations in the near native region which have energy values existing in deep wells on the energy landscape (see figure 39).

The heat capacities for 2D100a and 2D100b are very similar (as with all their observables) which is not surprising as they share the same length and have similar H ratios (0.55 and 0.56 respectively). For comparison these two curves are compared with those found by Wust and Landau [9] in figure 40.



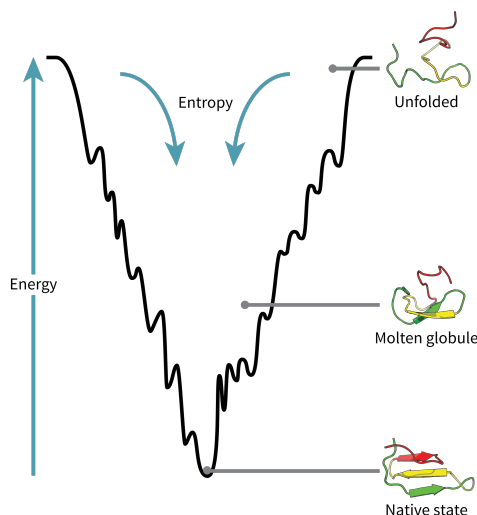


Figure 39: Diagram reflecting a rough (simplistic) 2D energy folding funnel of an arbitrary protein. At higher temperatures and energies the protein becomes denatured (unfolded) and entropy dominates. As  $T \rightarrow 0$  the protein forms a molten globule and then enters the near native region. The native state is located exactly at the minimum of the energy folding funnel. The transition temperature  $T_C$  can be located anywhere between the molten globule and native state.

The transition temperatures differed between the sequences but no accurate relationship could be deciphered between these temperatures and the chain lengths or hydrophobic ratios (see appendix D for table).

For the internal energy at low temperatures the observable continues to decrease and converge to a very small value. This is due to minimal thermal agitation and the compactness of structures in the native region.

Interestingly the free energy for sequences 2D64, 2D85, 2D100a and 2D100b all decrease with decreasing temperature past the critical temperatures, for example see figure 31. This contrasts with the free energies of 2D50 and 2D60 which show the free energy still gradually increasing (as with 2D50) or practically constant (as with 2D60). This could be due to the quality of convergence of the simulations. 2D50 and 2D60 converged very well and hence as explained in section 3.1 this relates to more accurate results. The free energies of 2D50 and 2D60 do align with expectations that the capacity to perform work increases with decreasing temperature, even in the  $T < T_C$  region.

The entropy of 2D60 reaches a peak just beyond  $T_C = 0.42$  and significantly drops in the low temperature region, this reflects the fast collapse of the protein chain into a compact native structure which has very low degeneracy. For 2D50 the entropy drops but not as rapidly as with 2D60, this could be due to differences in the folding funnel for the respective sequences. All the other 2D benchmark sequences, while qualitatively showing physically acceptable behaviour, had their entropies go below 0 as soon as the temperature passed from high-T through  $T_C$  to low-T. This cannot reflect the intrinsic physics since  $S/N < 0$  violates Boltzmann's entropy

formula:

$$S = k_B \cdot \ln[W] \quad (30)$$

where  $k_B$  is Boltzmann's constant and  $W$  is the number of microstates. It is unreasonable to conclude that the entire model used here is now considered redundant because of this violation, it is simply a matter of inadequate sampling for these sequences. More computational time and effort will lead to better convergence and more realistic observables.

In general the results for 2D benchmark (H)(P) sequences found here have unearthed the denatured  $\rightarrow$  globule  $\rightarrow$  native state transition as shown through the computation of thermodynamic observables. Whilst some observables 'broke' down in the low-T region beyond the critical temperature it can be corrected through more computational effort. The results of 2D60 are exemplary and characterise the thermodynamics of this sequence and lattice protein folding behaviour extremely well.

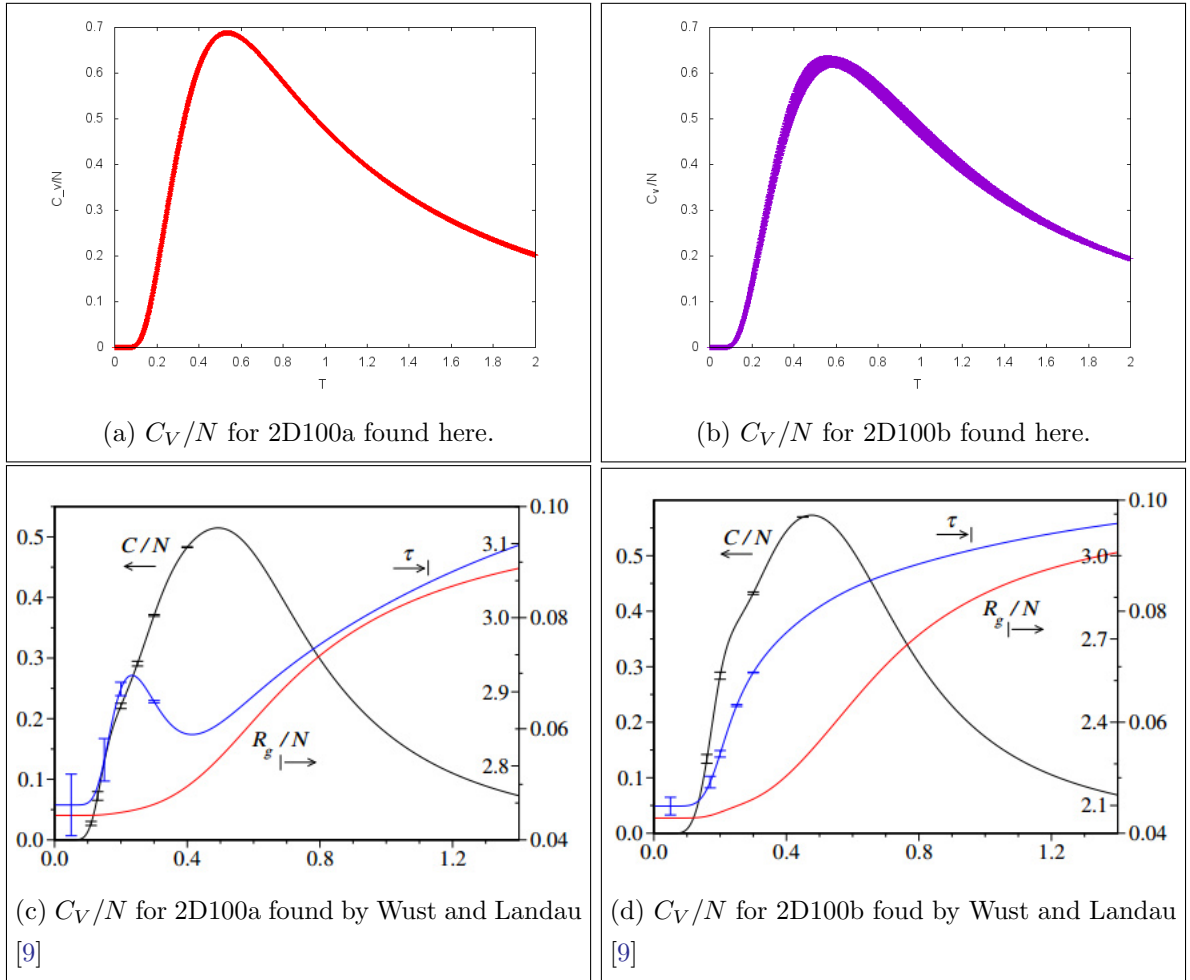


Figure 40: Notice the similarity in values and qualitative behaviour. One noticeable difference is the 'rough' quality the  $C_V/N$  from Wust and Landau has in the native region whereas my results are smoother. This could reflect that Wust and Landau ran their simulations for longer and hence sampled the conformational space in the native region more thoroughly.

## ISAWs

The specific heat capacity  $C_V/N$  for ISAWs of length 25, 64 and 100 shown in figure 35 shows only length specific behaviour in the temperature region of  $0.1 < T < 0.5$ . Beyond this range the heat capacity for the three lengths show universal behaviour. Stronger peaks can be seen with longer lengths of the homo-polymer, this is explained due to the fact that as  $N \rightarrow \infty$  the psuedo phase transition resembles a real phase transition. The growth in magnitude of the observable is due to the significant increase in H-H contacts.

The most accurate results obtained in this work was for the  $L = 25$  ISAW (see table 13), the modification factor reduction scheme seems to have taken on  $1/t$  functionality, due to the rapid coverage of conformational space of a short chain.

For lengths 64 and 100 the peak widths at the transition temperature are less than their (H)(P) sequence counterparts. This can be explained via the folding funnel, the globule  $\rightarrow$  native area on the folding funnel will be deeper and smoother for ISAWs than for protein (H)(P) chains.

The internal energy for ISAWs behaves similarly to that of protein sequences with a gradual decline from high- $T$  to  $T_C$  (with a faster rate closer to  $T_C$ ) then a convergence to a small value in the native region. Interestingly  $U/N$  is greater for the shortest ISAW in the native region but smallest in the  $T > T_C$  region.

A comparison that has not been made in the literature is between the internal energy found in WLS and with the genus/energy ratio in a simulation conducted to study the topology of pseudoknotted homopolymers [52]. The genus can be defined as the minimum number of handles the disk should have in order that all the cords are not intersecting<sup>15</sup> (see [52] for clarity).

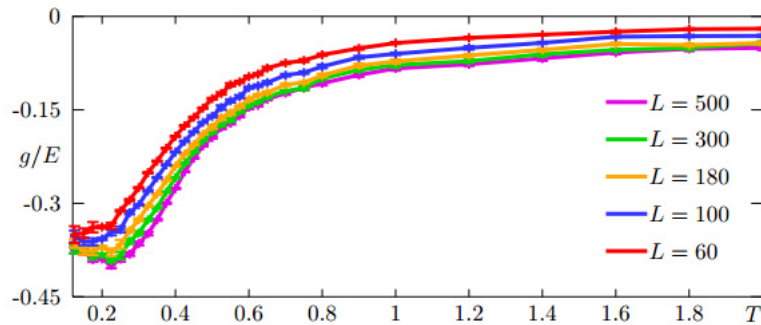


Figure 41: The ratio genus/energy of a homopolymer on a cubic lattice, as a function of  $T$ , at different lengths of the chain. Thanks go to the authors of [52].

One should compare the behaviour of  $g/E$  and  $U/N$  in figures 41 and 36 respectively, they share similar behaviour especially in the low- $T$  region. This comparison both reflects the tendency of the polymer chain to extend out into coil and 'string-like' structures with increasing

<sup>15</sup>Definition: The genus of a connected, orientable surface is an integer representing the maximum number of cuttings along non-intersecting closed simple curves without rendering the resultant manifold disconnected. [53]

temperature. I expect the genus will also share similar qualitative behaviour as the entropy also.

## 7 Conclusions

I have presented here a fully blind and straight forward parallel Monte Carlo scheme using Wang Landau sampling and the trajectory swapping method. Applying the trajectory swapping method to the problem of dynamical trapping and enhancing the efficiency of traversing conformational space to the HP model of lattice proteins has been successful. This has not been conducted within the existing literature. The development of the unique FRW trial move and its inclusion with pull, pivot, kink, pivot and bond re-bridging has enabled a different trial move set to complement WL sampling and to compare with the original work of Wust and Landau [9]. The trial move set is an important element of this Monte Carlo method and a huge amount of work was put in at the start of this project to develop original algorithms to implement trial moves which are described briefly in the literature. A thorough explanation and description of these moves has been presented to bridge the gap for those students and scientists willing to join the field of lattice polymer simulation.

Whilst WL is a powerful and generic Monte Carlo scheme to estimate thermodynamic behaviour its limitations were realized and an attempt to improve on the WL scheme consisted in implementing the  $1/t$  algorithm. In this scheme it is difficult to ensure that the modification reduction operation takes on  $1/t$  functionality, since the rate of coverage of conformational space differs for chain length and type. To ensure its regular success it will take committed tinkering of the Monte Carlo time for each sequence run.

Thermodynamic quantities were successfully computed for typical 2D benchmark sequences, some more accurate than others, which revealed the intrinsic folding and un-folding behaviour in the  $T > T_C$  and  $T \leq T_C$ . These computed observables also showed the existence of a denatured  $\rightarrow$  globule  $\rightarrow$  native structure pseudo phase transition. These results confirmed physical expectations and conclusions from related works. The amount of computational time and effort needed for longer sequences was under appreciated and in future it is fairly easy to obtain accurate results for these sequences (just run the simulations for a longer duration of time).

A successful native state search for 2D50, 2D60 and 2D64 was conducted. The native structure for 2D64 found here resonated with those found in other works and confirmed the expectation that protein sequences prefer to fold into particular shapes with a stable hydrophobic core. While the native states of 2D85, 2D100a and 2D100b was not attained this method came very close in a seemingly short amount of time. The difficulty in accessing these low temperature energy states for a simple lattice model emphasizes the challenge in protein structure prediction and sampling.

This Monte Carlo scheme was also applied to lattice homopolymers and their thermodynamic behaviour was also successfully investigated, a connection to a previously unrelated observable

(g/E) and 'classical' thermodynamic observables was made.

Overall I believe this project fulfils the aims that were set out in section 1.

## 8 Areas for Future Work

To begin simple with any new area and problem of science is essential. One first asks simple questions and with certain answers one can then ask more subtle questions to gather detailed knowledge and understanding of the problem at hand. This project is the simple beginnings in exploring the physics of protein folding and lattice polymer dynamics. There are many ways one can further expand on the work conducted here. I will mention but a few.

Firstly one could try and implement the  $1/t$  algorithm for the modification factor reduction successfully and try and find a way to code it such that it will adjust its definition of MC time so that the WL sampler will always converge asymptotically to the correct DOS. This is non-trivial and potentially time consuming, however very rewarding and ground breaking if it is done successfully.

A trivial expansion of this work is to increase the dimensions to 3 and investigate benchmark protein sequences. This would require the modification of the trial move algorithms and the lattice system. Though natural as this path is, there are more interesting pathways one could take since 3D benchmark sequences are already very well investigated. This however needs to be done at some point.

Within the same lattice dimensions and sequences used here one could study behaviour using variants of the Hamiltonian function (see equation 1). There are various HP matrices which could be investigated using this methodology for the first time (see [54] (Oct 2015) for further details). Also variants of the HOP model [47] could be generated and investigated and compare which HP energy function/matrix produces thermodynamic observables that best mimic the globular phase transitions seen in real proteins.

The most interesting expansion (or deviation?) of this work would be to develop a continuous model of the HP model using rotary degrees of freedom and simulate it using the recently outlined LLR (logarithmic linear routine) method for computing the DOS (see [55] for a detailed explanation and application). The trial move set used here could be assimilated into this continuous model. This would be novel work and it would be a great chance to compete with Wang-Landau sampling for the supreme algorithm for polymer and protein simulations.

## 9 Acknowledgements

For fruitful and useful conversations on general physics theory and computation I would like to thank Professor Simon Hands and Dr. Edward Bennett. I also thank the Department of physics for granting me access to Vivian room 606 which has been my workstation throughout this project and the use of the supercomputer which has been essential.

I would also like to mention Professor Adi Armoni and Dr. Maurizio Piai for discussions on my future in physics and for playing a part in my acceptance at UCL. This has helped with keeping me motivated and to enjoy the journey of research.

A huge thank you to my project supervisor Professor Biagio Lucini who has offered this unique opportunity to work on a problem in a growing field and for his expert guidance.

Many thanks to the regular MPHYS visitors of 606 who helped generate an stimulating and lively environment to work in.

# Appendices

## A Amino acid HP table

| <i>AMINO ACID</i>           | <i>CODE</i> | <i>H/P</i> |
|-----------------------------|-------------|------------|
| Alanine                     | A           | H          |
| Arginine                    | R           | P          |
| Asparagine                  | N           | P          |
| Aspartic Acid               | D           | P          |
| Asparagine or Aspartic Acid | B           | P          |
| Cysteine                    | C           | P          |
| Glutamine                   | Q           | P          |
| Glutamic Acid               | E           | P          |
| Glutamine or Glutamic Acid  | Z           | P          |
| Glycine                     | G           | P          |
| Histidine                   | H           | P          |
| Isoleucine                  | I           | H          |
| Leucine                     | L           | H          |
| Lysine                      | K           | P          |
| Methionine                  | M           | H          |
| Phenylalanine               | F           | H          |
| Proline                     | P           | H          |
| Serine                      | S           | P          |
| Threonine                   | T           | P          |
| Tryptophan                  | W           | H          |
| Tyrosine                    | Y           | P          |
| Valine                      | V           | H          |

## B Preliminary Testing Results

(These results were obtained during the initial procedures of the simulation development)

After testing the trial move sets and devising an energy computing routine I decided to see whether my program could produce the native configurations of chains with  $N_{monomers} < 20$ .

### B.1 Trial Move Prioritising

Since, as hinted at in section 3.5, pivot moves will have a smaller acceptance probability than performing a pull move (especially for longer chains in more compact configurations) so it was reasonable to impose that more pull moves were conducted on average than pivot moves. Pivot moves have the potential to drastically change the global configuration compared to pull and

kink flip moves, hence conducting a sufficient amount of this move will enable rapid coverage of configuration space (see [9] for their implementation too) which might outweigh the low acceptance rate. Kink flip moves are handy for performing tiny movements in configuration space since they change only one monomer. In these results pull, kink flip and pivot moves were conducted 60%, 15% and 25% of the time respectively.

The probabilities assigned to this exact procedure was arbitrary and at best simple guess work, however in the future a more systematized approach will be adopted to ensure an efficient simulation.

## B.2 Energy Scoring

The scoring system is simple: If the total energy of the configuration is less than the previous *known* minimum energy (which is = 0 to begin with) then set that as the new minimum energy. The configuration related to the minimum energy is then printed to file 'native.txt' which stores the coordinates of the monomers so that the configuration can be drawn either manually or via another program.

The computer routine for the following results is shown in figure 42. In all cases the chain starts out as a horizontal linear one.



```

//testing move section etc..
int energy=0, minenergy=0;
float ran,ki,pu;

for(mv=1;mv<=5000;mv++)
{ran=rndnum();
  //      printf("ran = %f\n", ran);

  if(ran<=1 && ran>0.4)
  {
    pullmove(N,ranseq(N),latlength,size);
    //  printf("PULL\n");
  }
  else if(ran<=0.15)
  {
    kinkflip(N,latlength,size);
    //printf("KINK \n");
    //printposarr(N);
  }
  else
  {
    pivot(N,ranseq(N),latlength,size);
    //printf("PIVOT \n");
    // printposarr(N);
  }

  energy = compenergy(N,latlength);
  if(energy < minenergy)
  {minenergy=energy;//set new minimum energy
    printf("NEW MIN ENERGY= %d\n",minenergy);//print to screen the new energy
    ofp=fopen("native","w");
    for(i=1;i<=N;i++)
    {
      fprintf(ofp,"%d is at POS= %d\n",i,POS[i]);//write coordinates to native.txt a new native config
    }
    fclose(ofp);
    energy=0;
  }
  else;
}
}

```

Figure 42: The routine in main which attempts 5000 moves on the chain recording minimum energy configurations.

### B.3 Results

In this section results are presented for short sequences of proteins which I have created for illustration purposes. The sequence name e.g. 2D7A represents the lattice dimension (2D),  $N_{monomers} = 7$  and a letter 'A' signifying its unique HP sequence. The results consist of the lowest minimum energy value and the chain diagram derived from the coordinates.

For all runs of the simulation the seed # will be specified.

#### 2D7A

2D7A has HP sequence: (HHPHHPH) and the results for 5 different seeds are presented in table 15.

| Seed # | $E_{min}$ | Configuration Type |
|--------|-----------|--------------------|
| 7412   | -2        | (i)                |
| 8293   | -2        | (ii)               |
| 2823   | -2        | (i)                |
| 6902   | -2        | (ii)               |
| 9382   | -2        | (iii)              |

Table 15: Configuration type (i), (ii) and (iii) are shown in figure 43.

The fact 3 distinct types of configuration were found only for 5 different seeds after 5000 attempted moves reflects the degeneracy of this short sequence with more (H) than (P) monomers.

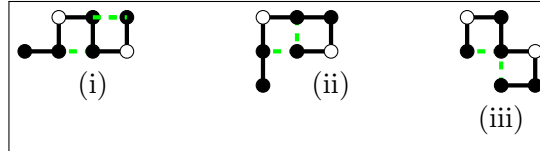


Figure 43: The configurations (i),(ii) and (iii) found in the 2D7A runs.

#### 2D10A

2D10A has HP sequence: (PHPHHPHPHH) and the results for 5 different seeds are presented in table 16. The number of attempted moves was 5000.

#### 2D20A

2D20A has HP sequence: (PHPHHPHPHHHPHPHHHPHH) (2 x 2D10A) and the results for 5 different seeds are presented in table 17. The number of attempted moves was 20000.

| Seed # | $E_{min}$ |
|--------|-----------|
| 9382   | -3        |
| 4510   | -3        |
| 6902   | -3        |
| 7523   | -3        |
| 7123   | -3        |

Table 16: Results for 2d10a

| Seed # | $E_{min}$ |
|--------|-----------|
| 7123   | -7        |
| 6521   | -7        |
| 8715   | -8        |
| 3829   | -8        |
| 5782   | -8        |

Table 17: Results for 2d20a

## 2D50A

A simulation run on a real benchmark (2D50) was attempted using 100000 attempted moves. The HP sequence for 2D50 is: (HHPHPHPHPHHHPHPPPHPPPHPPPHPPPHPPPHPH-HHHPHPHPHPH). The minimal energy found, using seed # 6138, was = -13. This however is not the minimum found in simulations using EMC, SISPER, EES and FRESS which found  $E_{min} = -21$  (See [31]).

## C Replica Exchange Routine

```
1 //===== REPLICA EXCHANGING =====
3 if (mv%1000==0)
4 {
5     for ( i=0; i<=(numprocs-1); i++)
6     { if (myid==0) //if I am master thread
7         { source=(int) (rndnum()*(numprocs-1)); //printf(" source= %d\n", source);
8           dest=i; //printf(" dest= %d\n", dest);
9         }
10    else;
11    MPI_Barrier(MPLCOMM_WORLD);
12    MPI_Bcast(&source, 1, MPI_INT, 0, MPLCOMM_WORLD);
13    MPI_Bcast(&dest, 1, MPI_INT, 0, MPLCOMM_WORLD);
14    if (source != dest)
15    {
16        if (myid==source)
17        {
18            MPI_Send(POS, possize, MPI_INT, dest, 1, MPLCOMM_WORLD);
19        }
20        else if (myid==dest)
21        {
22            MPI_Recv(POS, possize, MPI_INT, source, 1, MPLCOMM_WORLD,
23                MPI_STATUS_IGNORE);
24        }
25        else;
26    }
27    else;
28 }
29 }
30 else;
31 //=====
```

## D Critical temperatures for 2D benchmark sequences

| Sequence | $T_C$ | (H) ratio |
|----------|-------|-----------|
| 2D50     | 0.576 | 0.5       |
| 2D60     | 0.42  | 0.716     |
| 2D64     | 0.39  | 0.656     |
| 2D85     | 0.545 | 0.694     |
| 2D100a   | 0.535 | 0.55      |
| 2D100b   | 0.577 | 0.56      |

## 10 References

### References

- [1] Dawkins, R. (2006) *The Selfish gene (No. 199)*. **Oxford university press**
- [2] Alberts, B., Bray, D., Lewis, J., Raff, M., Roberts, K., Watson, J.D. and Grimstone, A.V. (1995) "*Molecular Biology of the Cell (3rd Editions)*". Trends in Biochemical Sciences 20(5).
- [3] Madras, Neal. and Slade, Gordon. (1996) *The Self-Avoiding Walk*. Birkhauser Boston. ISBN 0-8176-3891-1
- [4] Kerson Huang. *Lectures On Statistical Physics And Protein Folding*. World Scientific Publishing Co.Pte.Ltd. ISBN 978-981-256-150-3.
- [5] Dill, Ken.A., Bromberg, Sarina., Yue, Kaizhi., Fiebig, Klaus.M., Yee, David.P., Thomas, Paul.D. and Chan, Hue.Sun. (1995) **REVIEW** *Prcinciples of protein folding - A perspective from simple exact models*. Protein Science, 4:561-602. Cambridge University Press.
- [6] Landau, D. and Binder, K. (2009) *A Guide to Monte Carlo Simulations in Statistical Physics. 3rd. Edition*. Cambridge University Press. ISBN-13 978-0-521-76848-1 (HARDBACK).
- [7] Wang, Fugao. and Landau, David.P. (2000) *An efficient, multiple range walk algorithm to calculate the density of states* arXiv:cand-mat/0011174v1 [cond-mat.stat-mech]
- [8] D. P. Landau, Shan-Ho Tsai, and M. Exlerb (2004). Center for Simulational Physics, The University of Georgia, Athens, Georgia 30602. *A new approach to Monte Carlo simulations in statistical physics: Wang-Landau sampling* [DOI: 10.1119/1.1707017]
- [9] Wust, Thomas., Landau, David.P. (2012) *Optimized Wang-Landau sampling of lattice polymers: Ground state search and folding thermodynamics of HP model of proteins*. arXiv:1207.3974v1 [cond-mat.soft]
- [10] Van Kampen, N.G. (1998) *Remarks on Non-Markov Processes* Brazilian Journal of Physics, vol. 28, no. 2
- [11] Wust, T. and Landau, D.P. (2009) *Comput. Phys. Commun.* **179** 124 (2008)
- [12] Wust, Thomas., Li Ying.Wai. and Landau, David.P. (2013) *Unraveling the beautiful complexity of simple lattice model polymers and proteins using Wang-Landau sampling*. arXiv:1301.3466v1 [cond-mat.soft]
- [13] Anfinsen, C.B., Haber, E., Sela, M. and White, F.H. (1961) *Proc. Natl. Acad. Sci. USA*, **47**, 1309-1314.
- [14] Anfinsen, C.B.(1973) *Principles that govern the folding of protein chains* Science 181:223-30.
- [15] Dill, K.A. (1990) *Dominant forces in protein folding*. Biochemistry 29:7133-55.
- [16] Chen, J., Stites (2001) *Packing is a key selection factor in the evolution of protein hydrophobic cores*. Biochemistry 40:15280-89.
- [17] Dill, K.A., Ozkan, Banu S., Shell, M.Scott., and Weikl, T.R. (2008) *Annu. Rev. Biophys.* 37:289-316.
- [18] Piana, Stefano., Klepeis, John.L. and Shaw, David E. (2014) *Curr. Opin. Struct. Bio.* **24:98-105**
- [19] Howlett, David.R (2003) *Curr. Med. Chem - Immun., Endoc. & Metab. Agents* 3,371-383
- [20] Wolfenden, R. (2007) *Experimental measures of amino acid hydrophobicity and the topology of transmembrane and globular proteins*. J. Gen. Physiol. 129:357-62.

- [21] Hansmann, U.H.E. and Okamoto, Y. (1999) *Annual Reviews of Computational Physics VI*, ed. D. Stauffer. World Scientific Singapore.
- [22] Crescenzi, P., Goldman, D., Papadimitriou, C., Piccolboni, A. and Yannakakis, M. (1998) *On the complexity of protein folding* Journal of computational biology ,5(3).
- [23] Bahi, Jacques.M., Bienia, Wajciech., cote, Nathalie. and Guyeux, Christophe. (2013) *Is protein folding problem really a NP- complete one? First investigations* arXiv:1306.1372v1 [q-bio.BM]
- [24] Demaine, Erik.D. and O'Rourke, Joseph. (2007) *Geometric Folding Algorithms*. Cambridge University Press. ISBN 978-0-521-85757-4 (HARDBACK)
- [25] Clote, Peter. and Backofen, Rolf.(2000) *Computational Molecular Biology: An Introduction*. Wiley Ser. Math. Comp. Bio. ISBN 0-471-87251-2
- [26] Sugita, Yuji. and Okamoto, Yuko. (1999) *Replica-exchange molecular dynamics method for protein folding* Chemical Physics Letters 314 (1999) 141 -151
- [27] Tozzini, Valentina. (2005) *Coarse-grained models for proteins* Current Opinion in Structural Biology **15:144-150**
- [28] Lathrop, Richard.H. (1994) *The protein threading problem with sequence amino acid interaction preferences is NP-complete* Protein Engineering vol.7 no.9 10.59-1068. 1994
- [29] Peng, Jian. and Xu, Jinbo. (2011) *RaptorX: exploiting structure information for protein alignment by statistical inference* PROTEINS
- [30] Peng, Jian. and Xu, Jinbo. (2011) *A multiple-template approach to protein threading* PROTEINS
- [31] Zhang, Jinfeng. (2007) [www.people.fas.harvard.edu/~junliu/Workshops/talkSlides/JinfengZhang\\_MCW2007.pdf](http://www.people.fas.harvard.edu/~junliu/Workshops/talkSlides/JinfengZhang_MCW2007.pdf)
- [32] Robert, C.P. (2016) *The Metropolis-Hastings algorithm* arXiv:1504.01896v3 [stat.CO] 27 Jan 2016
- [33] Belardinelli, R.E. and Pereyra, V. D. (2007) *Fast algorithm to calculate density of states* Physical Review E 75, 046701
- [34] Belardinelli, R.E., Manzi, S. and Pereyra, V. D. (2008) *Analysis of the convergence of the 1/t and Wang-Landau algorithms in the calculation of multidimensional integrals* arXiv:0806.0268v1 [cond-mat.stat-mech]
- [35] Deutsch, J.M. (1996) *Long Range Moves for High Density Polymer Simulations* arXiv:cond-mat/9610116v1 [cond-mat:soft]
- [36] Zhang, Jinfeng., Kou, S.C. and Liu, Jun. S. (2007) *Biopolymer structure simulation and optimization via fragment regrowth Monte Carlo* THE JOURNAL OF CHEMICAL PHYSICS 126, 225101
- [37] Hartl, F.U., Bracher, A. and Hayer-Hartl, Manajit. (2011) *Molecular chaperones in protein folding and proteostasis* doi:10.1038/nature10317
- [38] Moore, Davis.S. and McCabe, George. P (1993) *Introduction to the practice of statistics* (2nd ed.) ISBN 0-7167-2250-X
- [39] F. Liang and W. Wong, J. Chem. Phys. 115, 3374 (2001)
- [40] J. L. Zhang and J. S. Liu, J. Chem. Phys. 117, 3492 (2002)
- [41] C. I. Chou, R. S. Han, S. P. Li, and T. K. Lee, Phys. Rev. E 67, 066704 (2003)

- [42] H. P. Hsu, V. Mehra, W. Nadler, and P. Grassberger, Phys. Rev. E 68, 021113 (2003)
- [43] S. C. Kou, J. Oh, and W. H. Wong, J. Chem. Phys. 124, 244903 (2006)
- [44] Koh, Y.W., Sim, Adelene Y.L. and Lee, H.K. (2015) *Dynamical traps in Wang-Landau sampling of continuous systems: Mechanism and solution* arXiv:1508.01888v1 [con-mat.stat-mech]
- [45] Vogel, T., Li, Y.W., Wust, T. and Landau, D.P. (2013) PHYS. REV. LETT. **110**, 210603
- [46] Shmygelska, Alena. and Hoos, Holger.H. (2005) *An ant colony optimisation algorithm for the 2D and 3D hydrophobic polar protein folding problem*. BMC Bioinformatics doi:10.1186/1471-2105-6-30
- [47] Shi, Go., Wust, T., Li, Y.W. and Landau, D.P. (2015) *Protein folding of the H0P model: A parallel Wang-Landau study* Journal of Physics: Conference Series **640**(2015)012017
- [48] Zhou, C. and Su, T. (2008) Phys. Rev. E **78**, 046705
- [49] Zhou, C. and Bhatt, R.N. (2005) Phys. Rev. E **72** 025701(R)
- [50] Walters, Peter. (1982) *An Introduction to Ergodic Theory* Springer, ISBN 0-387-95152-0
- [51] Wust, Thomas. and Landau, D. P. (2015) *Versatile approach to access the low temperature thermodynamics of lattice polymers and proteins* arXiv:1503.04433v1 [cond-mat.soft]
- [52] Vernizzi, Graziano., Ribeca, Paolo., Orland, Henri. and Zee, A. (2005) *The Topology of Pseudoknotted Homopolymers* arXiv:q-bio/0508042v2 [q-bio.BM]
- [53] Munkres, James. R. (2000) *Topology* Vol. 2. Upper Saddle River: Prentice Hall
- [54] Rashid, Mahmood. A., Khatib, Firas. and Sattar, Abdul (2015) *Protein preliminaries and structure prediction fundamentals for computer scientists* arXiv:1510.02775v1 [cs.CE]
- [55] Langfeld, K., Lucini, B., Pellegrini, R. and Rago, A. (Sept 2015) *An efficient algorithm for numerical computations of continuous densities of states* arXiv:1509.08391v1 [hep-lat]